IJNS

International Journal
of Network Security

ISSN 1816-353X (Print)
ISSN 1816-3548 (Online)

Vol. 25, No. 3 (May 2023)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan, R.O.C.
Email: mshwang@asia.edu.tw

International Journal of Network Security is published both in traditional paper form (ISSN 1816-353X) and in Internet (ISSN 1816-3548) at http://ijns.jalaxy.com.tw

PUBLISHER: Candy C. H. Lin

23-75, P.O. Box, Taichung, Taiwan 40199, R.O.C.

Volume: 25, No: 3 (May 1, 2023)

International Journal of Network Security

Industrial Internet Vulnerability Detection Method Based on CBAM-CNN-SVM

Peng-Shou Xie, Jia-Lu Wang, Hong Wang, Yin-Chang Pan, Xiao-Ye Li, and Tao Feng
(Corresponding author: Jia-Lu Wang)

School of Computer and Communications, Lanzhou University of Technology
No. 36 Peng Jia-ping road, Lanzhou, Gansu 730050, China
Email: wangjluu@163.com

(Received July 29, 2022; Revised and Accepted Mar. 28, 2023; First Online Apr. 13, 2023)

Abstract

The security vulnerabilities in the existing industrial Internet system differ from those in the traditional network. Using traditional vulnerability detection methods to detect security vulnerabilities in the industrial Internet is prone to problems of low detection accuracy and a high false negative rate. Not conducive to the safe operation of industrial Internet systems. Aiming at the above problems, this paper constructs the industrial Internet security vulnerability database through analysis, analyzes the characteristics of security vulnerabilities, and proposes a security vulnerability detection method for the industrial Internet based on CBAM-CNN-SVM. This method uses CBAM to optimize CNN, and the optimized CNN is used to extract vulnerability features. Finally, the extracted features are input into SVM to classify and detect vulnerabilities. The experimental results show that the method used in this paper can more accurately detect security vulnerabilities in multiple categories, and the accuracy is higher than the existing methods.

Keywords: Convolutional Block Attention Module; Convolutional Neural Network; Industrial Internet Security; Support Vector Machine; Vulnerability Detection

1 Introduction

At present, the combination of the industrial control industry and information technology is increasingly close, and the industrial Internet industry and high-tech technology are developing rapidly. Informatization has become an important driving force and technical support for the innovation and development of the traditional industrial control industry. However, with the prosperity and development of the industrial Internet, it is also faced with huge security threats [20]. During the development of industrial control systems, certain security measures and procedures are usually adopted, but as the running time continues to extend, the system will gradually expose different safety-related defects and vulnerabilities [4]. In addition, the exposure to the traditional industrial control system in the environment of the industrial Internet is huge, and the equipment, host, and industrial protocols in the industrial control system generally have the problem of old versions and models [14].

In actual production, in order to maintain the stability of the production environment, the existing vulnerabilities are often not repaired, and the software and hardware are not upgraded. A typical case is the "Eternal Blue" incident, which occurred mainly because the industrial host system was old and had not been patched in time, giving criminals an opportunity to cause serious loss of important data and property [18]. Statistics it shows that the industrial Internet production environment has become the hardest hit area of the "Eternal Blue" incident [15]. In order to effectively and timely discover the security vulnerabilities in the industrial Internet system, repair them in time, and improve the security of the overall industrial Internet environment, it is urgent to carry out systematic research work in the field of vulnerability detection [23].

In recent years, researchers have combined Natural Language Processing (NLP) and neural network models for software security vulnerability detection. Wartschinski *et al.* [21] proposed a vulnerability detection system based on deep learning of a natural code base, which alleviated the subjective influence of manual definition of vulnerability detection features by human experts, also reduced a lot of manpower, and improved the detection accuracy, showing that the machine can learn about the potential for vulnerability detection. Zou [28] and Li [12] applied deep learning methods to the field of vulnerability detection, which improved the accuracy of vulnerability detection. However, to solve the problem of data imbalance, manual annotation was adopted, which cost a lot of time and resulted in a high false positive rate. Ziems [27] and others for the first time use the most advanced transformer-based architecture to detect software security vulnerabilities, improve detection accuracy, and provide new ideas for subsequent researchers, but in the data preprocessing part, the excess files that may cause the model to be overfitted are manually deleted and processed, which increases the time overhead.

According to the purpose and granularity of vulnerability detection, the vulnerability identification technology is different. Dynamic detection methods [13], static detection methods [8], and the combination of dynamic and static detection methods are mainly used for detection [17]. Compared with dynamic detection, static detection is performed before the program runs, which can better analyze the source code and prevent the system from being damaged and causing immeasurable losses after the program runs. Therefore, this paper uses a static detection method to detect security vulnerabilities in the industrial Internet and carries out relevant research on vulnerability detection for source code. Static detection technology mainly includes graph-based learning [26], machine learning [25], natural language processing [27], and other methods. Although these methods improve the accuracy of vulnerability detection to a certain extent, there are still problems, such as a high false negative rate and high false positive rate. In addition, security vulnerability detection is mainly concentrated in traditional networks, and there is little research on the industrial Internet, which is not conducive to building a secure and stable industrial Internet environment.

To solve the above problems, this paper proposes an industrial Internet security vulnerability detection method based on CBAM-CNN-SVM, which integrates the attention mechanism and improves the Convolutional Neural Network(CNN). First, build an industrial Internet security vulnerability library, which is converted into a dataset after processing. In order to satisfy the data that can be input into the neural network for detection, and to reduce the impact of data imbalance on the experimental results, the dataset is processed. After that, the CNN was improved. First, a lightweight attention module was introduced, and then the last layer of CNN was replaced by a Support Vector Machine(SVM) to form the vulnerability detection model used in this paper. Finally, the detection of security vulnerabilities in the industrial Internet was realized.

2 Industrial Internet Security Vulnerability Detection Method

Efficient detection methods for industrial Internet security vulnerabilities are of great significance for maintaining the security of industrial Internet systems. This paper mainly describes the industrial Internet security vulnerability detection method used in the following two parts: the first part is the framework for describing the detection method as a whole, and the second part is a specific description of the technology used in the detection method.

2.1 A Framework Method To Vulnerability Detection

According to the research on the principles and methods of industrial Internet security vulnerability detection, this paper constructs the industrial Internet security vulnerability detection method as shown in Figure 1.

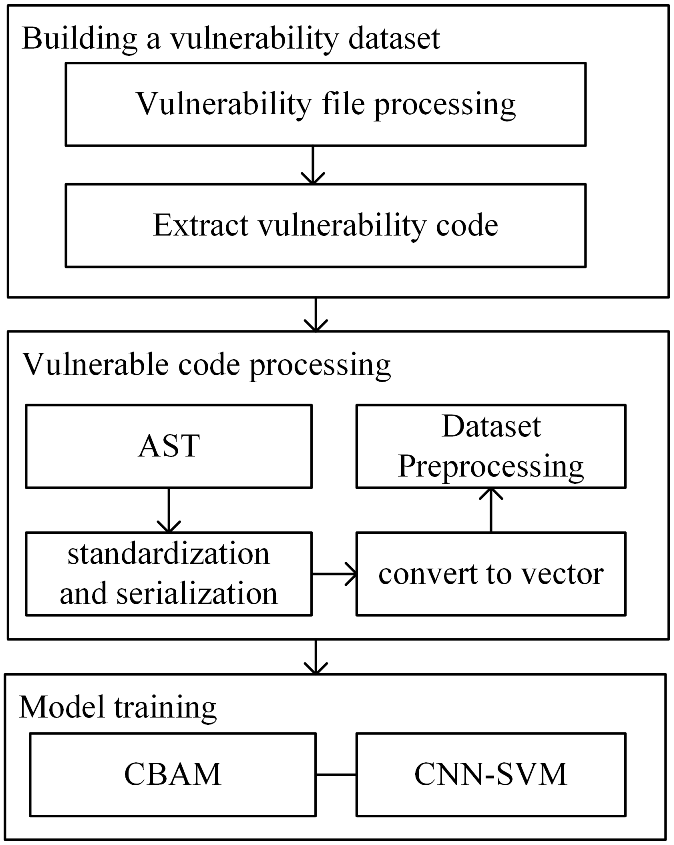

Building a vulnerability dataset
Vulnerability file processing
Extract vulnerability code
Vulnerable code processing
AST
Dataset Preprocessing
standardization and serialization
convert to vector
Model training
CBAM
CNN-SVM

Figure 1: Frame of industrial Internet security vulnerability detection method

Figure 1 shows that the framework consists of three parts: the construction of the industrial Internet security vulnerability database, the processing of industrial Internet security vulnerability data, and the training of the CBAM-CNN-SVM model.

The first part is to build the industrial Internet security vulnerability database. Due to the relatively closed industrial production environment, the equipment in industrial production is connected to the external environment through the network, which increases the risk of defect leakage [7]. Therefore, it is very necessary to carry out security vulnerability detection before the operation of industrial control equipment. Since there is no open and authoritative industrial Internet security Vulnerability Database, we first collected the types of security vulnerabilities in the industrial Internet through the China National Vulnerability Database (CNVD), matched the CVE numbers, extracted the Vulnerability files, and processed the Vulnerability files. The industrial Internet security vulnerability code is extracted to provide data support for subsequent research. The vulnerability data mainly comes from SARD and NVD.

The second part is to deal with the industrial Internet security vulnerability code. The industrial Internet security vulnerability database we built in the first part is the code snippet of C language. To detect the security vulnerabilities in the industrial Internet, the vulnerability code needs to be processed first. Firstly, the vulnerability code fragment is abstracted into Abstract Syntax Tree (AST), the obtained AST is standardized and serialized, and converted into vector form by the word embedding method. Then, the dataset of industrial Internet security vulnerability detection is formed after numerical processing, and then normalization is performed. Due to the large difference in the number of vulnerabilities among the constructed industrial Internet security vulnerability database, we balanced the dataset, and finally redivided the data to form a new dataset to meet the input of the vulnerability detection model.

The third part is to form an industrial Internet security vulnerability detection method, namely CBAM-CNN-SVM, by improving the CNN. To solve the problem that the number of parameters in the CNN model increases sharply with the deepening of the network level, the Resnet structure is selected in this paper. In order to improve the running speed, obtain more accurate classification results, and make the model focus on the features that contribute more to the classification results, we introduce a lightweight attention module, namely Convolutional Block Attention Module (CBAM). At the same time, considering the excellent classification ability and generalization ability of SVM, we combined CNN and SVM to form the CBAM-CNN-SVM industrial Internet security vulnerability detection method used in this paper. The dataset obtained in the second part is input into the constructed industrial Internet security vulnerability detection model, the semantic information about the vulnerability code is learned and the feature vectors are extracted to finally realize the detection and recognition of different categories of vulnerabilities in the industrial Internet.

2.2 Vulnerability Detection Method

The CBAM-CNN-SVM security vulnerability detection model constructed in this paper is shown in Figure 2.

Figure 2 shows the industrial Internet security vulnerability detection model, which inputs the preprocessed data into the constructed CNN, and adds a dual-channel attention mechanism, that is, CBAM, to the CNN, mainly including channel dimension attention mechanism and spatial dimension attention mechanism. The use of a dual-channel attention mechanism allows the CNN to focus more on the region relevant to the classification detection task. In addition, the last layer of the CNN is replaced with an SVM with better classification effect, and the final detection result is output.

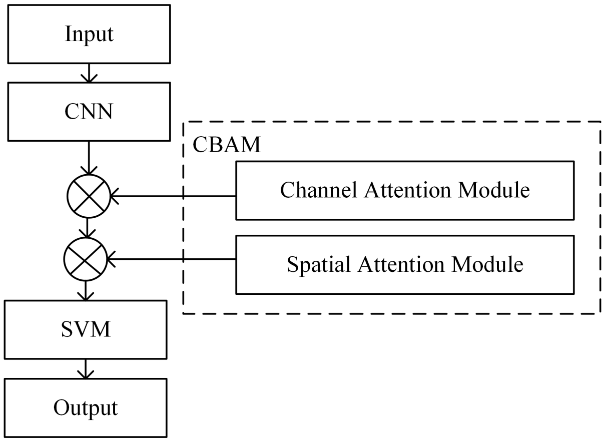


Figure 2: CBAM-CNN-SVM Industrial Internet security vulnerability detection model

```
void host_lookup(char *user_supplied_addr){
    struct hostent *hp;
    in_addr_t *addr;
    char hostname[64];
    in_addr_t inet_addr(const char *cp);
    validate_addr_form(user_supplied_addr);
    addr = inet_addr(user_supplied_addr);
    hp = gethostbyaddr(addr, sizeof(struct in_addr),
AF_INET);
    strcpy(hostname, hp->h_name);
}
```

Figure 3: Example code of C language for CWE-787

Based on the model shown in Figure 2, the first to build the industrial Internet security vulnerability database in the C language code snippet, turning it into a vector can input to the neural network model characteristics, after numerical value and normalized processing, and processed data input to build testing model, the vulnerability code inside the function dependency, The characteristics of different types of vulnerabilities are extracted and detected. This paper builds the CBAM - CNN - SVM based industrial Internet security vulnerabilities detection method, characteristics of different categories of loopholes as identification rules, after test set the input to the trained neural network model for testing, output the number of different categories and their corresponding holes for more classification of industrial Internet security vulnerabilities detection.

Take the industrial Internet security vulnerability numbered CVE-2019-5185 as an example, which is mainly caused by out-of-bounds writing. The CWE number is CWE-787, and its C language code fragment is shown in Figure 3.

This sample code represents taking the IP address from the user, verifying that it is properly formatted, then looking up the hostname based on the IP address, and copying the hostname into the buffer. As you can see, the function allocates a 64-byte buffer to store the hostname, but there is no guarantee that the user's host name will be smaller than 64 bytes. At this point, if the address specified by the attacker resolves to a very large host name, the function may overwrite sensitive data or even hand control flow to the attacker. In addition, the sample code contains an unchecked return value (CWE-252) that may cause a NULL pointer to be dereferenced (CWE-476).

Secondly, the processed data is input into CNN, which is first applied in the field of image classification and detection, and then extended to other fields of classification and detection [2], showing good performance in feature extraction. However, CNN also has some disadvantages, for example, with the increase in the number of layers of CNN, the accuracy of the model will be improved accordingly. However, when the number of layers is too many, the number of parameters will increase sharply, and the calculation amount will increase greatly. Too many parameters will also lead to overfitting and poor generalization ability of the model. To solve this problem, the residual network (ResNet) [10] structure is selected in this paper, and residual units are added through short-circuit mechanism, as shown in Figure 4.

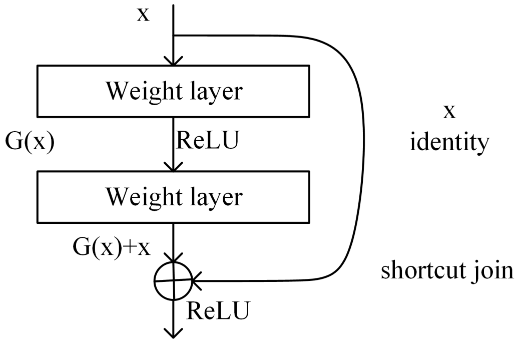


Figure 4: ResNet network structure

Where, x represents the input, $G(x)$ represents the output of the residual block before the activation function of the second layer, namely $G(x) = W_2\sigma(W_1x)$, where W_1 and W_2 represent the weights of the first and second layers, σ represents the ReLU activation function, and the output of the final residual block is $\sigma(G(x)+x)$, The curve on the right represents an identity mapping of input x, called a shortcut join. This residual structure realizes the jumping link in the model, and the input of one layer can directly cross several layers as the input of the following layer, which is helpful to solve the problems of a large number of parameters, gradient disappearance, and gradient explosion.

After that, the CBAM [22] is introduced. Compared with the traditional Attention mechanism that only focuses on a single dimension, CBAM can focus on the information of both channel and spatial dimensions, and has better performance. Its structure is shown in Figure 5.

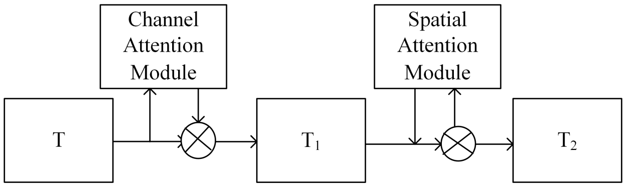


Figure 5: CBAM structure

Let the total feature of the input be $T \in R^{C*H*W}$, which means that C neurons can obtain an one-dimensional channel attention graph $M_C \in R^{C*1*1}$ and two-dimensional spatial attention graph $M_S \in R^{1*H*W}$ based on the pooling layer of width and height. The total process can be summarized as Equation (1) and Equation (2).

$$\begin{aligned} T_1 &= M_C(T) \otimes T && (1) \\ T_2 &= M_S(T_1) \otimes T_1 && (2) \end{aligned}$$

Here, $\otimes$ stands for element-level multiplication, and the broadcast mechanism is used for dimensional transformation and matching in the middle, T_1 stands for features under the action of the channel-dimension attention mechanism, and T_2 stands for features under the action of the spatial-dimension attention mechanism. Among them, the Channel Attention Module (CAM) in the Channel dimension assigns different weights to the classification results according to the influence of the features of security vulnerabilities on different channels, which can be expressed as Equation (3).

$$\begin{aligned} \mathrm{M_c(T)} &= \sigma(\ \mathrm{MLP\,(Avg\,Pool\,(T))}\ + MLP(MaxPool(T))) \\ &= \sigma\left(\mathrm{W_1}\left(\mathrm{W_0}\left(\mathrm{T^c_{avg}}\right)\right) + \mathrm{W_1}(\,\mathrm{W_0}\,(\ \mathrm{T^c_{max}}\ ))\right) \end{aligned} \qquad (3)$$

Where, σ is the sigmoid function, W_0 and W_1 are the shared weights of MLP, $W_0 \in R^{C*C/r}$, $W_1 \in R^{C*C/r}$.

Since not every feature in the input features of security vulnerabilities has the same contribution to the classification task, the Spatial Attention Module (SAM) in the spatial dimension assigns different weights to the input features according to their contribution, to achieve a more accurate classification effect, which can be expressed as Equation (4).

$$\begin{aligned} \mathrm{M_s(T)} &= \sigma\left(\mathrm{f}^{7*7}([\ \mathrm{Avg\ Pool\ (T);\ MaxPool\ (T)}\ ])\right) \\ &= \sigma\left(\mathrm{f}^{7*7}\left([\ \mathrm{T^s_{avg}}\ ;\mathrm{T^s_{max}}]\right)\right) \end{aligned} \qquad (4)$$

Where, σ is the sigmoid function, and f^{7*7} is the convolution kernel of $7*7$.

Finally, in this paper, the last layer of CNN is replaced by SVM as a classifier to achieve the final classification detection. The CNN in this method consists of multiple fully connected layers and is used as a feature extractor to learn local features and extract the most distinguishable features from the original dataset, and the output of each layer is used as the input of the next layer. SVM was first proposed by Cortes and Vapnik in 1995 [3]. The SVM adopted in this paper is essentially a linear binary classifier, which adopts the one-to-many method for multiple classifications. That is, for 9 categories of vulnerabilities that need to be classified, 9 SVMs are constructed, and each SVM distinguishes the data of this category from the data of other categories. The final output is determined by the SVM with the largest distance from the separation interface. For nonlinear sample data, SVM needs to introduce kernel function K to map low-dimensional nonlinear data into high-dimensional space, and construct a hyperplane to realize nonlinear data classification. In SVM, the training samples closest to the hyperplane and meeting certain conditions are called support vectors, as shown in Figure 6.

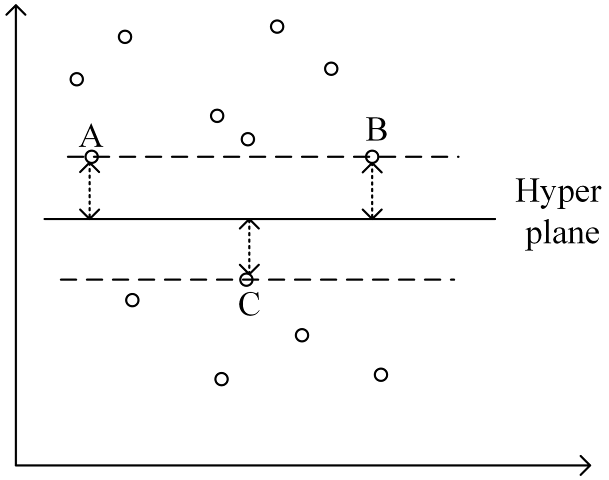


Figure 6: SVM schematic

Among them, the three points A, B, and C are called support vectors, and their distances to the hyperplane are equal and meet certain conditions. These three points jointly determine the hyperplane. As long as other points do not fall on or within the dotted line, the parameters of the hyperplane will not change. In this paper, the Sigmoid kernel function can be used to solve the global optimal value, which is conducive to the good generalization ability of the model for unknown samples and to prevent over-fitting.

Through the above methods, this paper constructs the industrial Internet security vulnerability detection method based on CBAM-CNN-SVM, realizes the detection and classification of security vulnerabilities, and further guarantees the security of the industrial Internet environment. The following experiments show that the proposed method has a better performance compared with other existing methods.

3 Experiment and Result Analysis

In order to prove that the method used in this paper has better results, the following three parts are described in detail, first introducing the experimental environment and evaluation indicators, then describing the detection process of industrial Internet security vulnerabilities, and finally analyzing the results of the experiment.

3.1 Experimental Environment and Evaluation Index

This paper adopts AMD Ryzen 5 4600U with Radeon Graphics 2.10ghz processor, RAM 16.0GB, Windows 10 home Chinese operating system, and programming environment is Python 3.8. Deep learning is based on the Keras 2.8.0 framework.

In this paper, a confusion matrix is used to evaluate the detection results, and there are four cases of evaluation results, namely TP, FP, TN, and FN, as shown in Table 1.

Table 1: Confusion matrix

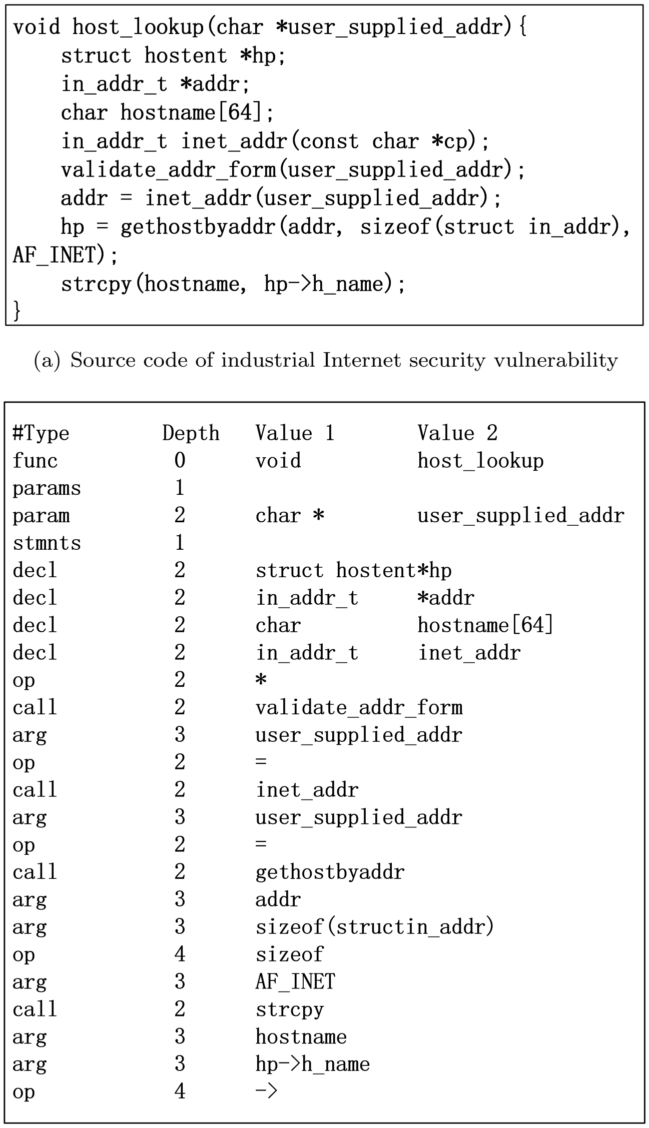
Evaluation indicator		Forecast result	
		correct classification	error classification
actual situation	correct classification	**TP**	**FN**
	error classification	**FP**	**TN**

Where TP and TN represent the correct number of vulnerability types classified; FP and FN indicate the number of errors in classifying vulnerability types. In addition, five widely used evaluation indicators are also used to evaluate the effectiveness of vulnerability detection, which are: False negative rate(FNR): the ratio of the number of samples with vulnerabilities but not detected to the total number of samples with vulnerabilities; False positive rate(FPR): that is, the ratio of the number of samples without vulnerability but detected as having vulnerability to the total number of samples without vulnerability; P(Precision): the ratio of the number of truly vulnerably detected samples to the total number of vulnerably detected samples; R(Recall) : the ratio of the number of truly vulnerable and correctly detected samples to the number of all vulnerable samples; ; Acc(Accuracy): The ratio of detecting the correct number of vulnerabilities to the total number of vulnerabilities detected is a common evaluation indicator, which is usually used to measure the quality of the classifier?F_1: Considering the overall effectiveness of both precision and recall, it can be expressed by Equation (5) as follows.

$$F_1 = \frac{2 * P * R}{P + R} \quad (5)$$

3.2 Industrial Internet Security Vulnerability Detection Process

The vulnerability code in the industrial Internet security vulnerability dataset is text data with certain grammatical rules. The industrial Internet security vulnerability



```
void host_lookup(char *user_supplied_addr){
    struct hostent *hp;
    in_addr_t *addr;
    char hostname[64];
    in_addr_t inet_addr(const char *cp);
    validate_addr_form(user_supplied_addr);
    addr = inet_addr(user_supplied_addr);
    hp = gethostbyaddr(addr, sizeof(struct in_addr),
AF_INET);
    strcpy(hostname, hp->h_name);
}
```

(a) Source code of industrial Internet security vulnerability

```
#Type        Depth   Value 1       Value 2
func          0      void          host_lookup
params        1
param         2      char *        user_supplied_addr
stmnts        1
decl          2      struct hostent*hp
decl          2      in_addr_t     *addr
decl          2      char          hostname[64]
decl          2      in_addr_t     inet_addr
op            2      *
call          2      validate_addr_form
arg           3      user_supplied_addr
op            2      =
call          2      inet_addr
arg           3      user_supplied_addr
op            2      =
call          2      gethostbyaddr
arg           3      addr
arg           3      sizeof(structin_addr)
op            4      sizeof
arg           3      AF_INET
call          2      strcpy
arg           3      hostname
arg           3      hp->h_name
op            4      ->
```

(b) C language code snippet converted to AST format

Figure 7: The source code of industrial Internet security vulnerabilities, and the serialized AST is listed in a tree structure

library constructed in this paper is composed of vulnerability code fragments in C/C++ format, which cannot be directly input into the constructed neural network model for detection. In order to retain the semantic information in the vulnerability code fragment, it is necessary to process the data first, and then traverse it to form an industrial Internet security vulnerability detection dataset, and then preprocess the dataset. In this paper, the "CodeSensor" parser is used to obtain the corresponding AST [5] from the source code of industrial Internet security vulnerabilities, and the serialized AST is listed in a tree structure, as shown in Figure 7(a) and Figure 7(b).

Among them, Figure 7(a) represents a piece of code involved in industrial Internet security vulnerabilities, Figure 7(b) represents the AST table obtained after its conversion, and #Type in the first column refers to the type name, for example, func means the function name, and params means the function parameter of the leaf node. The second column Depth indicates the Depth of the type; The remaining two columns, Value 1 and Value 2, indicate the name and Value corresponding to the function type. In order to improve the accuracy of model training, after the industrial Internet security vulnerability code is converted to AST, the AST is standardized, and then the Binary Encoder Representation from Transformers (BERT) word embedding method is used to map the node sequence to the vector table, forming the industrial Internet security vulnerability detection data set. In addition, in order to reduce the impact of data imbalance on the experiment, data sets need to be preprocessed, including character feature digitization, data balancing and normalization, as shown in Figure 8.

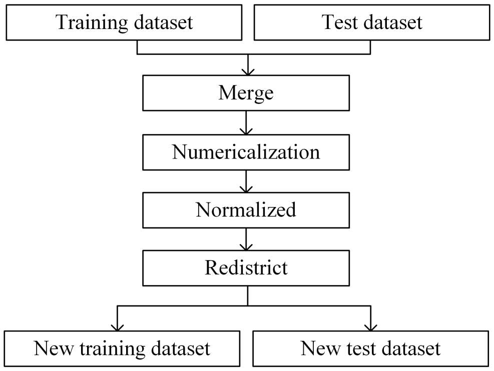


Figure 8: Example code of C language for CWE-787

Figure 8 is the preprocessing process of the data set. It processes the data set in the built industrial Internet security vulnerability database, divides it into new training sets and test sets, and then inputs them into the industrial Internet security vulnerability detection model built in this paper. The detection of industrial Internet security vulnerabilities can be summarized into two tasks, namely detection, and identification. Detection is mainly used to determine whether there are vulnerabilities, which is a binary classification problem, while identification is used to determine which type of detected vulnerabilities belongs, which is essentially a multi-classification problem.

3.3 Result Analysis

In combination with SARD and CWE vulnerability databases, an industrial Internet security vulnerability database is built, which contains 37153 vulnerability files. Due to the large difference in the number of different vulnerability types, in order to ensure that there is enough data to train the constructed vulnerability detection model and ensure the accuracy of the experimental results, we detect the top nine vulnerability types with the largest proportion, and number the vulnerability types as 0-8. The classification problem in neural networks is designed around the assumption that the number of each category is similar. If the number of different types of vulnerabilities differs greatly, the detection efficiency of the model for vulnerabilities will be low. To solve this problem and ensure an equal number of vulnerabilities in each category, random undersampling and random oversampling algorithms are adopted in this paper to balance the dataset [16], and finally, 30,696 vulnerability files and nine types of vulnerability are obtained. The comparison before and after balancing the dataset is shown in Figure 9.

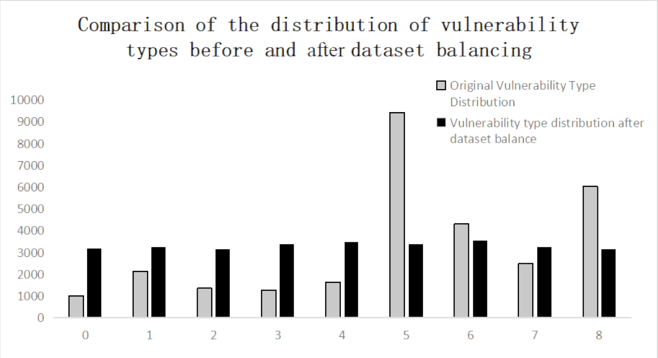
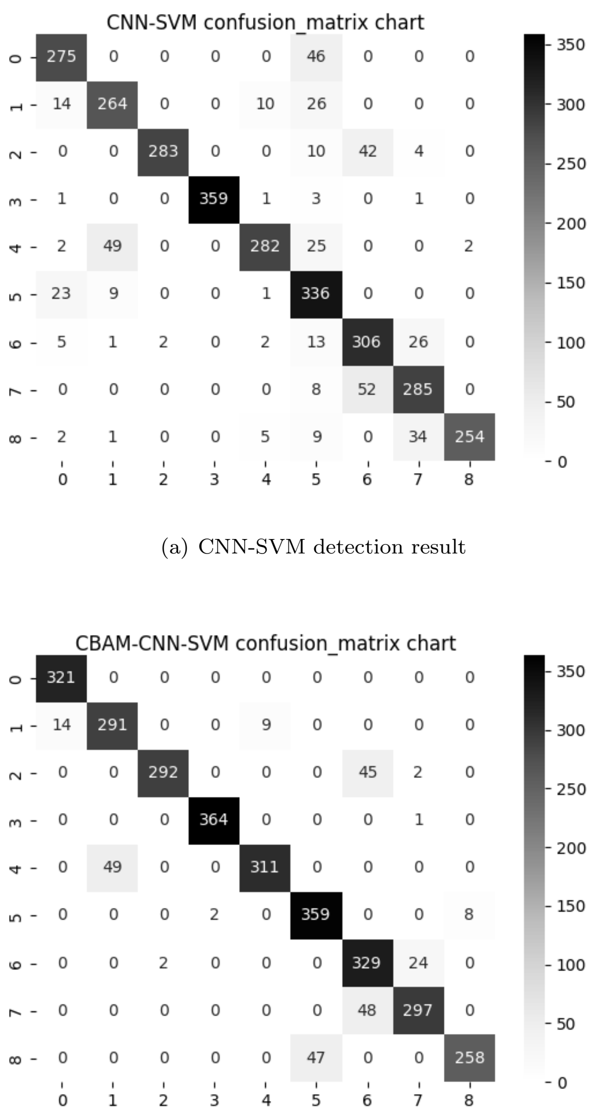


Figure 9: Comparison of vulnerability types before and after data balancing

Figure 9 shows that the distribution of vulnerability types in the processed dataset is relatively uniform, and the division ratio between the training set and the test set is 9:1, that is, the quantity ratio is 27623:3073. In order to prove that the method used in the experiment has better performance than the traditional network security vulnerability detection method, CNN-SVM and the improved CBAM-CNN-SVM are detected respectively, and the experimental results are shown in Figure 10(a) and Figure 10(b). On the premise of keeping the data processing method consistent, this paper first uses CNN-SVM method to detect the security vulnerabilities in the industrial Internet, and obtains the confusion matrix shown in Figure 10(a), with an accuracy of 86% and a high false-negative rate (FPR) of 14.62% in this method. To achieve better experimental results, we adopt the improved the CNN-SVM, namely the CBAM-CNN-SVM detection method, to perform multi-classification detection of vulnerabilities in the industrial Internet, and the results are shown in Figure 10(b).

Figure 10(b) shows that compared with the unimproved detection method CNN-SVM, the improved CBAM-CNN-SVM method has better overall performance, with the accuracy greatly improved to 91.8% and the false-negative rate reduced to 8.67%. It is proven that the improved method has good performance for security vulnerability detection in the industrial Internet. In order to continue to verify the superiority of this method in vulnerability detection, eight indicators such as TPR and TNR are compared and analyzed according to the obtained confusion matrix. In this paper, detection methods such as MLP [22], BiGRU [24], BLSTM [28], SVM [19], CNN [11], DCNN [9], CNN-SVM [1] and CNN-LSTM [6] are selected as comparative tests. Experimental results are quantified through these determined indicators, and the performance of the proposed detection method and other detection methods is comprehensively compared. The comparison results are shown in Figure 11.


CNN-SVM confusion_matrix chart

	0	1	2	3	4	5	6	7	8
0	275	0	0	0	0	46	0	0	0
1	14	264	0	0	10	26	0	0	0
2	0	0	283	0	0	10	42	4	0
3	1	0	0	359	1	3	0	1	0
4	2	49	0	0	282	25	0	0	2
5	23	9	0	0	1	336	0	0	0
6	5	1	2	0	2	13	306	26	0
7	0	0	0	0	0	8	52	285	0
8	2	1	0	0	5	9	0	34	254

(a) CNN-SVM detection result

CBAM-CNN-SVM confusion_matrix chart

	0	1	2	3	4	5	6	7	8
0	321	0	0	0	0	0	0	0	0
1	14	291	0	0	9	0	0	0	0
2	0	0	292	0	0	0	45	2	0
3	0	0	0	364	0	0	0	1	0
4	0	49	0	0	311	0	0	0	0
5	0	0	0	2	0	359	0	0	8
6	0	0	2	0	0	0	329	24	0
7	0	0	0	0	0	0	48	297	0
8	0	0	0	0	0	47	0	0	258

(b) Detection result of the proposed method

Figure 10: The experimental results

Figure 11 shows that the detection method adopted in this paper has high accuracy, high precision, and a low false negative rate. It can be seen that the detection performance of industrial Internet security vulnerabilities is better than other detection methods, and it can detect multiple types of vulnerabilities in the industrial Internet.

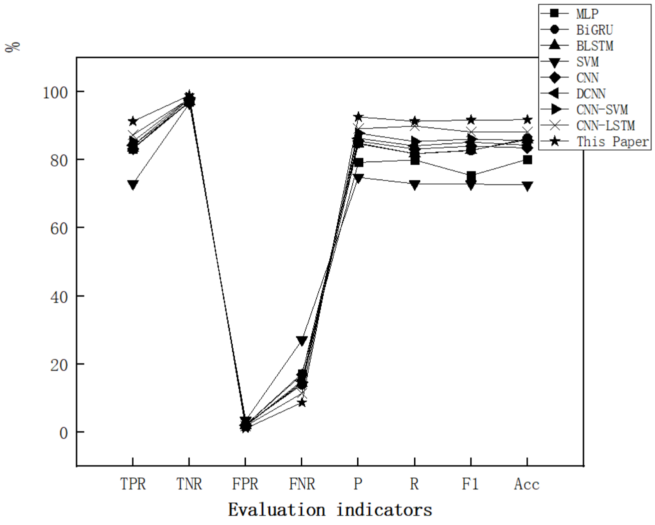


Figure 11: Comparison test results of different methods

4 Conclusions

In this paper, the industrial Internet security vulnerability database is constructed and an improved vulnerability detection method is proposed. Experimental results show that compared with other deep learning-based vulnerability detection methods, the industrial Internet security vulnerability detection method based on CBAM-CNN-SVM used in this paper can achieve higher detection accuracy and lower false negative rate, and has better detection performance. However, this paper still has some shortcomings and needs to be further improved. First of all, the detection granularity is still rough, and the function scale in the source code is different, so how to quickly locate the vulnerability points in different function scales and improve the accuracy of detection remains to be studied. Secondly, the product content of the industrial Internet security vulnerability dataset is lacking, and manual intervention is required after some data extraction, which limits the further development of industrial Internet security vulnerability detection. These are goals that we need to address in future research.

Acknowledgments

This research is supported by the National Natural Science Foundation of China under Grants No.61862040 and No.61762059. The authors gratefully acknowledge the anonymous reviewers for their helpful comments and suggestions.

References

[1] S. Ahlawat and A. Choudhary, "Hybrid cnn-svm classifier for handwritten digit recognition," *Procedia Computer Science*, vol. 167, pp. 2554–2560, 2020.

[2] S. Chakraborty, R. Krishna, and Y. Ding, "Deep learning based vulnerability detection: Are we there yet," *IEEE Transactions on Software Engineering*, 2021.

[3] V. Chauhan, K. Dahiya, and A. Sharma, "Problem formulations and solvers in linear svm: a review," *Artificial Intelligence Review*, vol. 52, no. 2, pp. 803–855, 2019.

[4] O. Emmanuel and J. Onochie, "Enterprise cyber security challenges to medium and large firms: An analysis," *International Journal of Electronics and Information Engineering*, vol. 13, no. 2, pp. 77–85, 2021.

[5] H. Fen, X. Fu, H. Sun, H. Wang, and Y. Zhang, "Efficient vulnerability detection based on abstract syntax tree and deep learning," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 722–727, Toronto,Canada, July 2020. IEEE.

[6] J. Guo, Z.Wang, H. Li, and Y. Xue, "Detecting vulnerability in source code using cnn and lstm network," *Soft Computing*, pp. 1–11, 2021.

[7] J. He, J. Yang, K. Ren, W. Zhang, and G. Li, "Network security threat detection under big data by using machine learning," *International Journal of Network Security*, vol. 21, no. 5, pp. 768–773, 2019.

[8] A. Kaur and R. Nayyar, "A comparative study of static code analysis tools for vulnerability detection in c/c++ and java source code," *Procedia Computer Science*, vol. 171, pp. 2023–2029, 2020.

[9] H. Kumawat, M. Chakraborty, and A. Raj, "Diatradsatnet—a novel lightweight dcnn architecture for micro-doppler-based small unmanned aerial vehicle (suav) targets' detection and classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–11, 2022.

[10] K. Lee, "Deep residual network notes," 2021.

[11] Y. Lee, H. Kwon, S. Choi, S. Lim, S. Baek, and K. Park, "Instruction2vec: Efficient preprocessor of assembly code to detect software weakness with cnn," *Applied Sciences*, vol. 9, no. 19, p. 4086, 2019.

[12] X. Li, L. Wang, and Y. Xin, "Automated software vulnerability detection based on hybrid neural network," *Applied Sciences*, vol. 11, no. 7, p. 3201, 2021.

[13] Y. Li, L. Ma, and L. Shen, "Open source software security vulnerability detection based on dynamic behavior features," *Plos one*, vol. 14, no. 8, p. e0221530, 2019.

[14] C. Lin, L. Huang, Y. Chen, and M. Hwang, "Research on security and performance of blockchain with innovation architecture technology," *International Journal of Network Security*, vol. 23, no. 1, pp. 1–8, 2021.

[15] P. Lin and Y. Chen, "Network security situation assessment based on text simhash in big data environment," *International Journal of Electronics and Information Engineering*, vol. 21, no. 4, pp. 699–708, 2019.

[16] S. Liu, G. Lin, Q. Han, S. Wen, J. Zhang, and Y. Xiang, "Deepbalance: Deep-learning and fuzzy oversampling for vulnerability detection," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 7, pp. 1329–1343, 2019.

[17] F. Mateo, J. Bermejo, J. Sicilia, and I. Argyros, "On combining static, dynamic and interactive analysis security testing tools to improve owasp top ten security vulnerability detection in web applications," *Applied Sciences*, vol. 10, no. 24, p. 9119, 2020.

[18] S. Saket and K. Kakelli, "A review on deep-learning based network intrusion detection systems," *International Journal of Electronics and Information Engineering*, vol. 13, no. 4, pp. 170–179, 2021.

[19] F. Wang and X. Wang, "Detect cross-site scripting attacks using average word embedding and support vector machine," *International Journal of Network Security*, vol. 24, no. 1, pp. 20–28, 2022.

[20] S. Wang, W. Li, and C. Kong, "Analysis of rear-end collision accident of urban traffic based on safety prewarning algorithm," *International Journal of Network Security*, vol. 23, no. 1, pp. 180–185, 2021.

[21] L. Wartschinski, Y. Noller, and T. Vogel, "Vudenc: Vulnerability detection with deep learning on a natural codebase for python," *Information and Software Technology*, vol. 144, p. 106809, 2022.

[22] S. Woo, J. Park, and J. Y. Lee, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, Munich, Germany, Sep. 2018.

[23] Y. Xue, "Research on network security intrusion detection with an extreme learning machine algorithm," *International Journal of Network Security*, vol. 24, no. 1, pp. 29–35, 2022.

[24] H. Yan, S. Luo, and L. Pan, "Han-bsvd: a hierarchical attention network for binary software vulnerability detection," *Computers & Security*, vol. 108, p. 102286, 2021.

[25] W. Zheng, J. Gao, and X. Wu, "The impact factors on the performance of machine learning-based vulnerability detection: A comparative study," *Journal of Systems and Software*, vol. 168, p. 110659, 2020.

[26] Y. Zhuang, Z. Liu, and P. Qian, "Smart contract vulnerability detection using graph neural network.," in *IJCAI*, pp. 3283–3290, Yokohama,Japan, Jul 2020.

[27] N. Ziems and S. Wu, "Security vulnerability detection using deep learning natural language processing," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1–6, Vancouver, Canada, May 2021. IEEE.

[28] D. Zou, S. Wang, S. Xu, Z. Li, and H. Jin, "μ vuldeepecker: A deep learning-based system for multiclass vulnerability detection," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2224–2236, 2019.

Biography

Peng-shou Xie was born in Jan. 1972. He is a professor and a supervisor of master student at Lanzhou University of Technology. His major research field is Security on Internet of Things. E-mail: xiepsh_lut@163.com.

Jia-lu Wang was born in Jan. 1998. She is a master student at Lanzhou University of Technology. His major research field is network and information security. E-mail: wangjluu@163.com.

Hong Wang was born in Mar. 1995. He is a master student at Lanzhou University of Technology. Her major research field is network and information security. E-mail: 2967589625@qq.com.

Yin-Chang Pan was born in Mar. 1993. He is a master student at Lanzhou University of Technology. His major research field is network and information security. E-mail:1713974116 @qq.com.

Xiao-Ye Li was born in Oct. 1995. She is a master student at Lanzhou University of Technology. His major research field is network and information security. E-mail: 976339400 @qq.com.

Tao Feng was born in Dec. 1970. He is a professor and a supervisor of Doctoral student at Lanzhou University of Technology. His major research field is modern cryptography theory, network and information security technology. E-mail: fengt@lut.cn.

High Quality Image Steganography Model Based on Encoder-Decoder Networks and 2D Logistic Chaotic Encryption

Qiu-Yu Zhang, Xue-Wen Hu, and Zhen Wang
(Corresponding author: Qiu-Yu Zhang)

School of Computer and communication, Lanzhou University of Technology
No. 287, Lan-Gong-Ping Road, Lanzhou 730050, China
Email: zhangqylz@163.com, luthuxw@163.com

(Received July 25, 2022; Revised and Accepted Mar. 9, 2022; First Online Apr. 13, 2022)

Abstract

Aiming at the problems that the existing image information hiding schemes based on generative adversarial network (GAN) cannot avoid the attackers from discovering the secret image between the residual image of the stego image and the cover image, and the color distortion occurs in the stego image when the color image is used as the cover image, high-quality image steganography model based on encoder-decoder networks and 2D logistic chaotic encryption was proposed, which consists of three convolutional neural networks: the encoding network, decoding network and discriminative network. The secret image is encrypted by a chaos encryption algorithm, then hidden in the Y channel of the cover image. The encoder-decoder networks model based on GAN is used to obtain a visually better stego image by minimizing the probability distribution of the stego image and the cover image. The discriminator analyzes the generated stego image and provides parameters to the encoder for optimizing the encoding network. The experimental results show that the proposed model can generate high-quality stego images, effectively prevent the leakage of the secret image information and color distortion of the stego image, and ensure the security of the secret image transmission.

Keywords: Chaotic Encryption; Encoder-Decoder Networks; Generative Adversarial Network; Image Hiding; Steganography

1 Introduction

Image steganography [5] is one of the information hiding techniques, which uses the statistical redundancy of image data and the human perception to hide meaningful secret information into the cover image, make it impossible for unauthorized persons to judge whether the original secret image is hidden in the cover image in order to achieve the purpose of covert communication [24]. Traditional image information hiding techniques are divided into spatial domain and transform domain information hiding techniques, the cover image's embedding capacity is low and the visual effect of the image is not ideal, and it is difficult to resist the evolving steganalysis technology, personal privacy is increasingly threatened, and the security of secret information is not easy to guarantee [5, 23, 24].

In recent years, the deep learning networks have been used widely in the field of information hiding [36]. Deep learning networks can not only improve the security and capacity of steganographic models, but also improve the detection accuracy of steganalysis models. After Goodfellow *et al.* [12] proposed GAN model, Hayes *et al.* [13] applied GAN to image information hiding for the first time, compared with traditional information hiding techniques, this method largely enhanced the embedding capacity and security of the secret information. Volkhonskiy *et al.* [28] proposed the use of deep convolutional GAN in image steganography, which made the application of steganography become more secure and reliable, provided a new research direction for existing image steganography and offered an opportunity to combine information hiding with deep learning networks.

The encoder-decoder networks based on GAN has been a widely used steganographic model in the field of existing image data hiding. However, this type of technology has certain defects. When the original color image is used as the cover image, the stego image will appear color distortion, and when an image is hidden in another image of the same size, the residual image of the two is more obvious, the attacker can find the secret image information from it. Some studies have proposed the method of "encrypt first, then embed" to ensure the security of the secret images [26, 31], but the encryption effect was difficult to guaranteed, and the security of the secret images still needed to be improved. In addition, the GAN models have been used widely in the image field and have achieved good results, after the encrypted secret image enters the network model, the quality of the extracted the secret image will be reduced.

Therefore, in order to prevent attackers from discovering the secret image information between the residual image of stego image and cover image, improve the security of the secret image transmission and eliminate the color distortion of the stego images, high quality image steganography model based on encoder-decoder networks and 2D logistic chaotic encryption is proposed. The main contributions of this paper are as follows:

1) The 2D logistic chaotic mapping image encryption is combined with the GAN encoder-decoder network, the secret image is encrypted before being hidden so that even if the third party obtains the stego image, it is still impossible to successfully extract the secret image, which can enhance security of the secret images.

2) The Xu-Net [29] steganalysis tool is introduced and improved as the discriminator of the basic network structure. Comparing with the original structure, the improved network uses the spatial pyramid pooling layer, it can prevent the problem of overfitting and avoid the loss of image information, so that its performance in steganalysis is better.

3) In order to solve the problem that there are noise and poor quality of the image after extracting the secret image, the noise reduction method is used to denoise the secret image, so that the differences between the adjacent pixels of the secret image become smaller.

The rest of the paper is organized as follows: Section 2 deals with the related research works. Section 3 is the description of the proposed steganography scheme. Section 4 gives the experimental results and analysis, and compares the performance with different existing steganography schemes. At last, there is the conclusion in Section 5.

2 Related Works

The existing image information hiding techniques [16] are divided into traditional image information hiding [1, 9] and image information hiding based on deep learning in the field of multimedia information security. At present, image information hiding techniques based on deep learning are roughly divided into convolutional neural network (CNN)-based and GAN-based image information hiding techniques. For example, Kich *et al.* [14] proposed an encoder-decoder networks based on CNN, which hided a color image in another color image of the same size with a large data hiding capacity, but the security of the secret image was not considered. Liu *et al.* [22] proposed a hiding scheme based on U-Net and wavelet transform, which combined the advantages of U-Net in image detail feature processing with the ability of wavelet transform to segment image details. The quality of the generated and extracted images was good, but the robustness of the images was poor.

Lin *et al.* [20] proposed an image steganography framework for neural style transfer based on Y channel information and a novel structural loss, this method strengthened robustness and expanded the application scenarios of image steganography, but the embedding capacity was low. Duan *et al.* [8] proposed a difference image grafting deep hiding (DIGDH) model that combined the overfitting characteristics of deep neural networks with the difference image grafting symmetrically method, which can generate images with higher similarity to the original images. However, the generated images were difficult to resist advanced steganalysis tools, and the quality of stego images were poor in the visual effect of the human visual system (HVS).

The above CNN-based encoder-decoder networks have a large steganographic capacity, but have poor visual effects, some steganography models use the idea of adversarial training, take steganography and steganalysis network as "opposites", and both of them conduct adversarial training to improve the anti-steganalysis detection ability of the steganography model [11,28]. For example, Liu *et al.* [21] to use semantic image synthesis technique to obtain stego images directly, with high accuracy in information extraction but significant distortion of image content. Qin *et al.* [25] proposed coverless image steganography based on GAN to generate stego images, optimize the quality of image by adversarial framework, but due to the limitation of the coverless method itself, the embedding capacity was low. Li *et al.* [17] proposed a cross feedback mechanism that significantly improved the capability of resisting steganalysis by simulating the competition between generator and a discriminator in the reconstructed GAN to learn automatically the embedding costs. But this model was not experimented in the domain of color images.

Encoder-decoder networks model of image steganography uses embedded carrier-based approach, which can improve the quality of the stego image to a certain extent compared with other steganography algorithms and models. For example, Zhu *et al.* [35] proposed a hiding data with deep networks (HiDDeN) model for encoder-decoder networks based on GAN, which was trained to generate a robust network by adding a noise layer between the encoder and decoder, and the generated image had a good visual effect, but image distortion still occurred when faced with image processing such as geometric transformation and contrast change. Zhang *et al.* [33] proposed an end-to-end scheme using SteganoGAN that supported the cover images of different sizes and arbitrary binary data, avoid partial steganalysis detection tools while increasing embedding capacity. Zhang *et al.* [34] proposed an invisible steganography via GAN (ISGAN) scheme, which can hide grayscale images in color images, and less color differences were generated between the original cover images and the stego images, but the stego images were less robust. Fu *et al.* [10] proposed a steganographic model HI-GAN, encoding network was consisted of residual blocks, this network framework hided a color secret image into another color image of the same size, which can output an image with lower distortion and higher visual quality.

Yang *et al.* [30] proposed an enhanced GAN-based technology using UT-6HPF-GAN, which designed a double-tanh function to approximate the optimal embedding simulator and achieved embedding and extracting message by syndrome-trellis code (STC). This scheme shortened the training time, but the security of images was low. Li *et al.* [18] proposed an adversarial image steganography with adversarial networks (AdvSGAN) model to perform encoding-decoding conversion in adversarial neural networks to achieve steganographic operations, while deceiving the target steganalysis tools, which had better security in the case of high-capacity embedding. Yuan *et al.* [32] proposed an end-to-end image steganographic scheme based on GAN, namely ADF-IS (attack and deep fusion for image steganography), which achieved pixel-level depth fusion and ensured high payload information embedding, but generated stego images with poor robustness.

Shi *et al.* [27] proposed a hidden message cycle GAN (HCGAN), which had high decoding accuracy and generated stego image with better quality. The above GAN-based steganography algorithms may have the risk that the attacker obtains the cover image and illegally obtains the secret image from the cover image and the stego image. To solve such problems, Yang *et al.* [31] proposed a high-capacity image steganography algorithm that combined image encryption and deep learning. Even if the attacker obtained the cover image, the secret information cannot be extracted, which improved the security of the stego image. Ding *et al.* [7] proposed a Cycle-GAN-based encryption and decryption network (DLEDNet) to implement the hiding process of medical images, which can protect medical images with higher security level, while encrypting and decrypting images in a more efficient way. Alkhelaiwi *et al.* [2] proposed to apply a homomorphic Paillier encryption scheme to a custom convolutional neural network, which not only ensured the security of the data, but also guaranteed a good classification accuracy.

In summary, in order to make the confidentiality and imperceptibility of the secret image be more effective, the secret image is firstly been pre-processed of encryption, and then it is hidden. This paper adopts an encoder-decoder structure based on GAN, the color image is used as the original cover image and the grayscale image as the original secret image. It is necessary to separate firstly the color channels of the cover image, then encrypt the secret image, embed the encrypted secret image into the Y channel of the separated cover image through the encoding network, which better solves the problem of color distortion and low security of the stego image.

3 The Proposed Scheme

Figure 1 shows the framework of the high quality image steganography model based on encoder-decoder networks and 2D logistic chaotic encryption. The framework is composed of three convolutional neural networks, include the encoding network, decoding network and discriminative network. Under the condition of high embedding capacity and imperceptibility, the generated stego image has high visual quality and security.

As shown in Figure 1, the proposed model can hide an arbitrary grayscale secret image in a color cover image of the same size by introducing an encoder-decoder networks model of the GAN, which uses the steganalysis tool as the discriminative network to calculate the difference between the cover image and the stego image. When the difference is small enough, the generated stego image has better effect on the HVS system. In addition, the proposed model uses the 2D logistic chaotic encryption method to encrypt the secret image before it enters the encoding network, so that it remains in the encrypted state in the network, thereby enhancing the concealment and security of the stego image. The encrypted secret image is extracted by the decoding network and then processed by noise reduction, and the secret image can be recovered with high quality.

3.1 Encryption and Decryption of Secret Image

3.1.1 Pixel position Scrambling Encryption Algorithm Based on 2D Logistic Mapping

In order to ensure the security of the secret image in the model of Figure 1, so that it is impossible for criminal to extract the secret image after obtaining the cover image. In this paper, the secret image is encrypted by 2D logistic chaotic mapping before entering the network. Logistic mapping is a classical nonlinear chaotic dynamical system. On this basis, people have studied the 2D logistic chaotic map, and dynamic behavior of the 2D logistic chaotic map is controlled by its parameters μ, λ_1, λ_2, γ. The system is in a chaotic state when parameters μ=4, λ_1=λ_2∈$[0.65, 0.9]$, γ=0.1. In this paper, through the double encryption method, the security of the secret image can be guaranteed even after decoding.

Suppose that grayscale secret image is s and its corresponding encrypted secret image is s'. Its size is assumed to be $M \times N$, the specific encryption algorithm steps are as follows:

Step 1. Input a secret image s of size $M \times N$.

Step 2. Initialize logistic control parameters, set μ=4, λ_1=λ_2=0.9, γ=0.1. At this time, the obtained 2D logistic system is in a chaotic state. The initial number of iteration K is set to 50, set key key_0=$\{x_0, y_0\}$, where x_0 and y_0 are the initial values of the chaotic map. Iterate through logistic system with key key_0=$\{x_0, y_0\}$ for K times. As shown

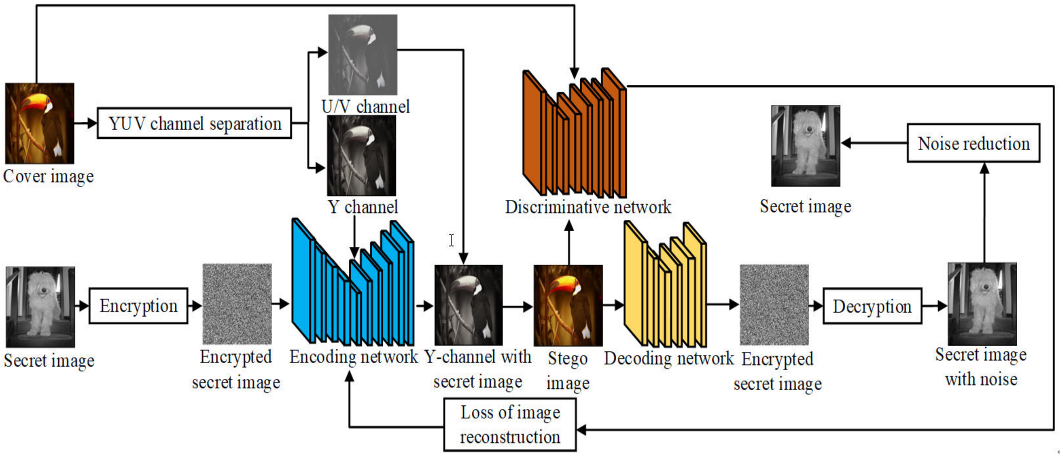


Figure 1: The framework of the proposed encoding-decoding network model

in Equation (1), obtain K pairs of chaotic sequence values.

$$\begin{cases} x_{n+1} = \mu\lambda_1 x_n(1-x_n) + \gamma y_n \\ y_{n+1} = \mu\lambda_2 x_n(1-y_n) + \gamma x_n \end{cases} \quad (1)$$

Step 3. Store the obtained K pairs of chaotic sequence values in 1D sequences $\boldsymbol{P}$ and $\boldsymbol{Q}$ of $M \times N$ respectively, do the modulo operation as in Equation (2) for the elements in $\boldsymbol{P}$ and $\boldsymbol{Q}$, get two integer arrays $\boldsymbol{P}'$ and $\boldsymbol{Q}'$

$$f(t_i) = mod(\lfloor t_i \times 10^{15} \rfloor, 10^6) \quad (2)$$

where $t_i \in \boldsymbol{P}$, $\boldsymbol{Q}$, t_p and t_q represent the elements in the sequences $\boldsymbol{P}$ and $\boldsymbol{Q}$ respectively.

Step 4. Generate two 1D pseudorandom sequences of length $M \times N$ are $\boldsymbol{P}''$ and $\boldsymbol{Q}''$ by sorting the arrays $\boldsymbol{P}'$ and $\boldsymbol{Q}'$ respectively, whose element values are unequal integers within $[0, M \times N - 1]$.

Step 5. The elements $t''_p(k)$ and $t''_q(k)$ in the 1D pseudorandom sequences $\boldsymbol{P}''$ and $\boldsymbol{Q}''$ are transformed as shown in Equation (3), map them to 2D scrambled matrix $\mathbf{X}$ and $\mathbf{Y}$, whose size are $M \times N$.

$$\begin{cases} x(i,j) = mod(t''_p(k), M) \\ y(i,j) = mod(t''_q(k), N) \\ i = k/N, j = k/M \\ k = 0, 1, ...M \times N - 1 \end{cases} \quad (3)$$

where the symbol "/" indicates the quotient operation, $x(i,j)$ and $y(i,j)$ are the elements of the 2D scrambled matrices $\mathbf{X}$ and $\mathbf{Y}$ respectively.

Step 6. Use the scrambling matrices $\mathbf{X}$ and $\mathbf{Y}$ to scram the secret image s. Transform each pixel value in the image s matrix to the corresponding position according to the values of the scrambled matrices $\mathbf{X}$ and $\mathbf{Y}$, to obtain the scrambled encrypted image s'.

3.1.2 Decryption Algorithm

The decryption process is the inverse process of the encryption process, it is necessary to acquire the given chaotic system key. There are mainly the following steps:

Step 1. Obtain the same key key_0=$\{x_0, y_0\}$ as the encryption process, and generate the scramble matrix $\mathbf{X}$ and $\mathbf{Y}$ from the key key_0.

Step 2. The encrypted image s' which was scrambled is inversely scrambled by the scramble matrices $\mathbf{X}$ and $\mathbf{Y}$, and the decrypted secret image s is obtained.

3.2 Image Hiding and Extraction

3.2.1 Hiding Network

The encoding network (hiding network) of the proposed scheme hides the secret image into the cover image, and uses an RGB cover image with a size of 256×256×3 and a grayscale image with a size of 256×256 as the input of the hiding network, the cover image is channel-separated before entering the network, and the encrypted secret information is embedded into the Y channel of the cover image by encoding network, so that the color of the generated stego image is not distorted. Figure 2 shows the specific framework of the hiding network, and its configuration details are shown in Table 1.

The structure of the hiding network consists of 5 downsampling and 5 upsampling, which are implemented by convolution (Conv) operation and deconvolution (DeConv) operation respectively, and the output feature maps by each network layer have the same size. The first layer is a convolutional structure with convolution kernel size of 3×3 and *stride*=1. The ninth layer is a deconvolutional structure with convolution kernel size of 3×3 and *stride*=1. Smaller convolution kernels can capture more details of the image during subsampling. Both network layers use batch normalization (BN) and the LeakyReLU activation function. The LeakyReLU activation function

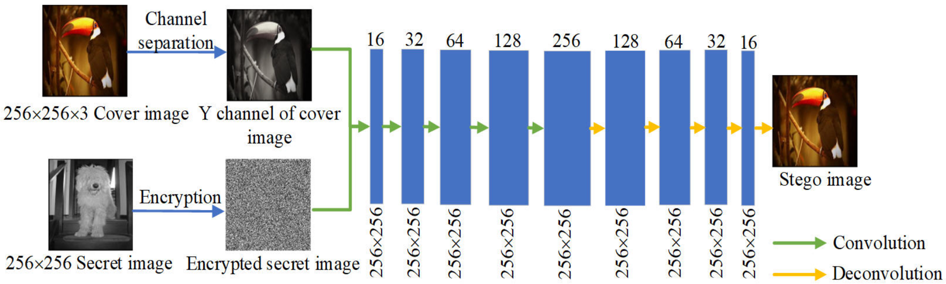


Figure 2: The framework of hiding network

Table 1: Configuration details of hiding network

Layers	Process	Input Size	Output Size
Input	/	/	2×256×256
Layer1	16×3×3 Conv+BN+LeakyReLu	2×256×256	16×256×256
Layer2	Inception	16×256×256	32×256×256
Layer3	Inception	32×256×256	64×256×256
Layer4	Inception	64×256×256	128×256×256
Layer5	Inception	128×256×256	256×256×256
Layer6	Inception	256×256×256	128×256×256
Layer7	Inception	128×256×256	64×256×256
Layer8	Inception	64×256×256	32×256×256
Layer9	16×3×3 DeConv+BN+LeakyReLu	32×256×256	16×256×256
Layer10	1×1×1 DeConv+Tanh	16×256×256	1×256×256

is the variant of the ReLU activation function, which can solve the problem of gradient disappearance when the number of neural network layers is large, and speeds up the convergence speed of the network. The expression of LeakyReLU activation function is shown in Equation (4).

$$LeakyReLU(x) = \begin{cases} x, x > 0 \\ \alpha x, x \leq 0 \end{cases} \in R, \alpha \in (0,1) \qquad (4)$$

The use of the Tanh activation function in the tenth layer solves the problem of slow convergence in most network layers when using the sigmoid() function, and basically does not produce a gradient explosion. The expression of Tanh activation function is shown in Equation (5).

$$Tanh(x) = \frac{\sinh(x)}{\cosh(x)} = 2sigmoid(2x) - 1 \qquad (5)$$

The Inception layer adopts the InceptionV1 network layer, and the InceptionV1 network layer contains convolution kernels of different sizes, which are 1×1, 3×3, 5×5 respectively. Because very deep networks are more prone to overfitting, they are difficult to transmit gradient updates to the entire network. Stacking large convolution layers together is computationally expensive. The InceptionV1 network layers stack the convolution operations and the pooling operations, which increase the width of the network on the one hand and the adaptability to the scale of the network on the other hand. The use of the InceptionV1 layers can perform multiple convolution operations or pooling operations on the input image in parallel, and stitch all the results into a very deep feature map. Different convolution operations and pooling operations can obtain different information of input images, process these operations in parallel and combine all the results to obtain better image features.

3.2.2 Extraction Network

The decoding network (extraction network) of the proposed scheme has a 6-layer convolutional network, whose input is an RGB stego image of size 256×256×3, and the output is a grayscale image of size 256×256. The extraction network consists of 3 downsampling and 3 upsampling, which are implemented by convolution and deconvolution operations. Except for the sixth layer, each other layer uses the BN operation and the LeakyReLU activation function, which speeds up the network training while the model's non-linear fitting ability is guaranteed, restoring the hidden secret image to the greatest extent. Each convolution operation uses convolution kernel of 3×3 with *stride*=1, except for the last layer where the convolution kernel size is 1×1. Figure 3 shows the specific framework of the extraction network, and its configuration details are shown in Table 2.

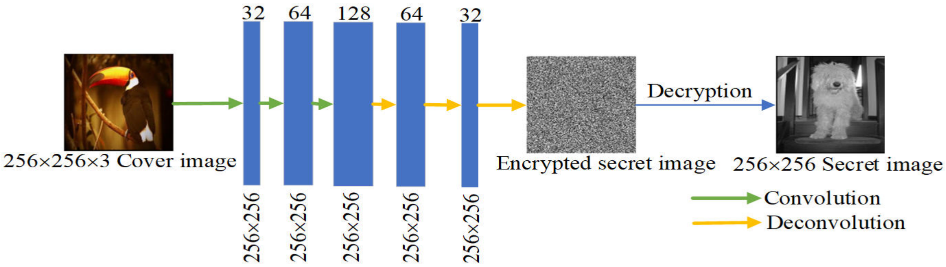


Figure 3: The framework of extraction network

Table 2: Configuration details of extraction network

Layers	**Process**	**Input Size**	**Output Size**
Input	/	/	3×256×256
Layer1	32×3×3 Conv+BN+LeakyReLu	3×256×256	32×256×256
Layer2	64×3×3 Conv+BN+LeakyReLu	32×256×256	64×256×256
Layer3	128×3×3 Conv+BN+LeakyReLu	64×256×256	128×256×256
Layer4	64×3×3 DeConv+BN+LeakyReLu	128×256×256	64×256×256
Layer5	32×3×3 DeConv+BN+LeakyReLu	64×256×256	32×256×256
Layer6	1×1×1 DeConv+Sigmoid	32×256×256	1×256×256

3.3 Discriminative Network

The adversarial network can be trained by competing steganography and steganalysis against each other. In general, the stronger discriminator has better detection performance. Through a series of adversarial training can achieve more secure steganography. The discriminative network in proposed scheme improves the Xu-Net [29] steganalysis tool, and makes use of its good steganalysis performance to generate the stego images with more anti-steganalysis. Figure 4 shows the specific framework of the discriminative network, and its configuration details are shown in Table 3.

In order to better express the distortion measurement, a series of parameters need to be trained to improve the effectiveness of the discriminator, discriminative network uses the structure of Conv-BN-LeakyReLU-Average Pooling (Avgpool) as the basic processing block. The discriminator all uses smaller convolution kernels, which can achieve better steganalysis performance. Using kernels of different sizes in the convolution process can eventually learn an accurate distortion probability distribution, thereby guiding the encoding network to achieve safer steganography.

In this paper, the GAN idea is introduced into the encoder-decoder network structure, where the encoding network acts as a generator and the decoding network completes the task of extracting secret image, and the CNN-based steganalysis tool is used as discriminator in the whole network. The Xu-Net steganalysis tool proposed by Xu *et al.* [29] is adopted in this paper, which replaces the global average pooling layer of the original Xu-Net with a spatial pyramid pooling layer, it can convert image convolutional features of any scale into the same dimension, cropping and scaling in the convolutional process are avoided, thereby image information is not lost to the greatest extent. In addition, comparing to the original structure which used a high-pass filter (HPF) with the same kernel, whose parameters are not optimized during the training process and the Tanh activation function was used in the pre-network layer. In order to adapt to the encoder-decoder network itself, this paper improves the Xu-Net steganalysis tool, which alleviates the problem of gradient disappearance and over-fitting caused by the introduction of Tanh activation function in the deep network. Instead of using Tanh activation function in the first and second layers of Xu-Net steganalysis tool, proposed scheme uses LeakyReLU activation function and the output of some neurons is zero, which make the network be sparsity and reduce the dependence between computation and parameters, alleviating the occurrence of the overfitting problem. In this paper, Tanh activation function is introduced in the last fully connected layer, so that the deep neural network is no longer linear combination of inputs, but can approximate any function, which increases the nonlinearity of the neural network. In addition, by adding discriminator into the steganography scheme, the generated stego image can be more similar to the original cover image. The expression of the loss function of the discriminator is shown in Equation (6).

$$Loss_{dis} = E_{c\sim p_c}[\log D(c) + E_{c\sim p_c, s\sim p_s}[\log(1 - D(G(c,s)))]] \quad (6)$$

where E represents the expected value, P_c and P_s represent the distribution of the cover image c and the secret image s respectively, D represents the discriminative network and G represents the generator.

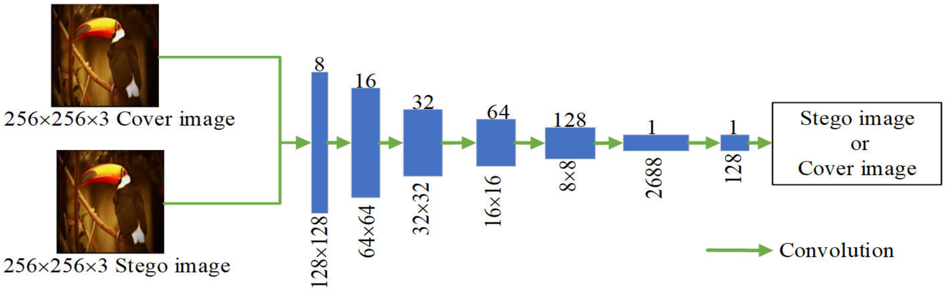


Figure 4: The framework of discriminative network

Table 3: Configuration details of discriminative network

Layers	Process	Input Size	Output Size
Input	/	/	3×256×256
Layer1	8×3×3 Conv+BN+LeakyReLu+Avgpool	3×256×256	8×128×128
Layer2	16×3×3 Conv+BN+LeakyReLu+Avgpool	8×128×128	16×64×64
Layer3	32×1×1 Conv+BN+LeakyReLu+Avgpool	16×64×64	32×32×32
Layer4	64×1×1 Conv+BN+LeakyReLu+Avgpool	32×32×32	64×16×16
Layer5	128×3×3 Conv+BN+LeakyReLu+Avgpool	64×16×16	128×8×8
Layer6	Spatial Pyramid Pool	128×8×8	2688×1
Layer7	Fully Connected Layer	2688×1	128×1
Layer8	Fully Connected Layer+Tanh	128×1	2×1

3.4 Loss Function

First define the hiding part and the extraction part, the hiding part is the sender who uses key_0 as the key, encrypt the secret image s which uses the encryption algorithm $E(x, y)$ of Section 3.1.1, then input the encrypted secret image s' and the cover image c into the hiding network $S_{encode}(x, y)$ to get the stego image s_e, finally the stego image s_e is sent to the receiver. The hiding part is shown in Equation (7).

$$s_e = S_{encode}(E(s, key_0), c) \tag{7}$$

The receiver inputs the received stego image s_e into the extraction network $S_{decode}(x, y)$, get the encrypted secret image s', finally use the decryption algorithm $R(x, y)$ of Section 3.1.2 to decrypt s' and obtain secret image s'' with noise points. The extraction part is shown in Equation (8).

$$\begin{cases} s' = S_{decode}(s_e) \\ s'' = R(key_0, s') \end{cases} \tag{8}$$

Evaluation criteria of traditional image data hiding schemes include PSNR, MSE, etc. for quantifying the difference between a cover image and a stego image, or between a secret image and an extracted secret image. In order to minimize the loss values of the stego and cover images, extracted secret and secret images, it is necessary to optimize the hiding network and the extraction network to minimize the error of image reconstruction. Therefore, this paper uses mean square error (MSE) as loss function of model to measure the difference between images. The MSE is shown in Equation (9).

$$MSE(y_i, y_i') = \frac{\sum_{i=1}^{n}(y_i - y_i')^2}{n} \tag{9}$$

where n is the number of samples, y_i is the expected output value, y_i' is the predicted value given by the neural network and MSE is the mean error function.

The loss function of the hiding network consists of three main components: the difference between the cover image c and the stego image s_e, the difference between the secret image s' and the encrypted secret image s' obtained by the extraction network and the auxiliary role of the discriminator in the image generation phase. Introduce $\omega_i(i{=}1, 2, 3)$ as the weight parameter for steganography, extraction and discrimination tasks respectively, to balance the weights between the three tasks. The expression for the hiding network is shown in Equation (10).

$$\begin{aligned} Loss_{encode} &= \omega_1 MSE(c, s_e) + \\ &\omega_2 MSE(s'', S_{decode}(s_e)) + \omega_3 Loss_{dis} \end{aligned} \tag{10}$$

The extraction network only needs to consider the accurate extraction of the secret image, and its loss function is shown in Equation (11).

$$Loss_{decode} = MSE(s'', S_{decode}(s_e)) \tag{11}$$

3.5 Denoise Processing

Because the secret image is encrypted by 2D logistic chaotic mapping, and the secret image s' needs to be decrypted, which leads to the appearance of noise points in the secret image s''. At the end of the network structure in Figure 1, this paper uses the total variation noise reduction (TV) model [4] to denoise the image s'' to obtain the secret image s. From the visual effect, this method not only removes the original noise from the image, but also effectively preserves the edge information of the image. The expression of TV noise reduction model is shown in Equation (12).

$$TV(u) = \int\limits_{\Omega} \mid \nabla u \mid dxdy \tag{12}$$

where $u(x, y)$ is the original noise-free image, Ω is the image definition domain and ∇u is the image gradient.

According to the basic idea of the TV noise reduction model in Equation (12), only the image u corresponding to the minimum $TV(u)$ is obtained, the image u is the image after noise reduction.

3.6 Secret Communication Algorithms

The sender and receiver follow the algorithms below for secret communication.

Information hiding algorithm. The sender uses the cover image c and the secret image s as the driver, input into the trained hiding network S_{encode}, through continuous adjustment of the discriminative network D, generate the stego image s_e, and then transmit the s_e.The detailed algorithm is shown in **Algorithm** 1.

Information extraction algorithm. After the receiver acquires stego image s_e, s_e is extracted by extraction network S_{decode} and then decrypted, obtain the secret image s'' with noise points. The detailed algorithm is shown in **Algorithm** 2.

Algorithm 1 The algorithm of stego image generation

```
Input: cover image $c$, secret image $s$, encryption key $key_0$
Parameter: loss adjustment $\omega_1$, $\omega_2$, $\omega_3$, discriminative
    network loss $Loss_{dis}$, training epochs $t\_epoch$
Output: stego image $s_e$
train the model: obtain $S_{encode}$, $S_{decode}$ and $D$
$s'$=Encrypt($s, key_0$)
for epoch in 1...$t\_epoch$ do
    $s_e = S_{encode}(s', c)$
    Minimize: Encoder-Decoder loss $Loss_{encode}$=
    $\omega_1 MSE(c, s_e)$+$\omega_2 MSE(s'', S_{decode}(s_e))$+$\omega_3 Loss_{dis}$
    Update $S_{encode}$, $S_{decode}$ and $D$
    Learning rate decay
end for
return $s_e$
```

Algorithm 2 The algorithm of secret image extraction

```
Input: stego image $s_e$, decryption key $key_0$
Parameter: training epochs $t\_epoch$
Output: secret image $s$
train the model: obtain $S_{decode}$
for epoch in 1...$t\_epoch$ do
    $s' = S_{decode}(s_e)$
    $s'' = Decrypt(s', key_0)$
    Minimize: Encoder-Decoder loss $Loss_{decode}$=
    $MSE(s'', S_{decode}(s_e))$
    Update $S_{decode}$
    Learning rate decay
    $s$=Denoise($s''$)
end for
return $s$
```

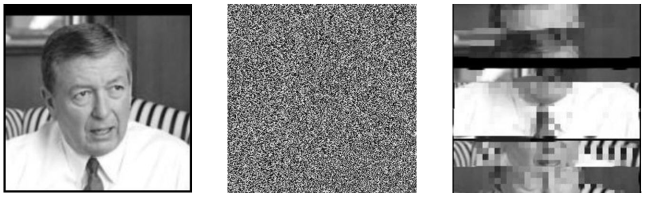
4 Experimental Results and Performance Analysis

This paper used a total of 13,000 face images integrated from LFW dataset [15], 7000 images and 4000 images were used for training and testing, respectively. The batch size of training was 8, the number of iterations epoch=5000, the initial network learning rate was 0.001 and the Adam algorithm was used to optimize basic model. All experiments were performed in TensorFlow on the platform of an Inter(R) Xeon(R) Sliver 4210 CPU @2.20GHz and a NVIDIA and a NVIDIA GeForce RTX 3080 Ti. The device memory was 36GB (2933MHz) and the application was Python 3.9. This paper also evaluated the performance of the scheme using two other classical image sources, ImageNet [6] and COCO [19], the images of the datasets were resized to 256×256 before entering the network.

4.1 Security Analysis

4.1.1 Comparison of Encryption Methods

Figure 5 shows the comparison result of the encrypted image between the proposed scheme and the Yang's scheme [31], it can be seen intuitively that the encrypted images are better scrambled in the proposed scheme.



(a) Original (b) Ours (c) Yang's

Figure 5: Image encryption results comparison

Figure 6 shows the comparison and analysis of the histogram of the encryption results of the proposed scheme and the Yang's scheme [31], it can be seen that the histogram distribution of the encrypted image of the proposed scheme is uniform, and the encrypted image displays little statistical information, which indicates the image encryption scheme of this paper is resistant to histogram statistics and has high security.

The size of the images to be encrypted were 256×256 in this paper, the encryption algorithm of Yang's scheme [31] was to exchange the position of the pixel block with the size of 16×16. The encryption scheme of this paper used 2D logistic chaotic encryption mapping to scramble the pixels in the image, eliminating the correlation between pixels by transferring the gray values of the pixel points of the image to be encrypted into random coordinates, so that the original image became a disturbed image. The results of the proposed scheme were visually more disorganized, messier and more secure than the encrypted images generated by Yang's scheme [31].

4.1.2 Residual Image Visibility Analysis

This paper proposed an end-to-end training model to replace the earlier step-by-step training model, the size of input images to the encoding network is $256 \times 256 \times 3$ during the training process. As shown in Figure 7, 4 images were selected from the LFW dataset as cover images, which entered the encoding network to generate the corresponding stego images, and the residual images were visualized by calculating the pixel difference between the original cover images and the stego images. The modification of the cover images after the secret images were embedded in the cover image were measured from visual perspective. After that, the residual images were enlarged by 5 times and 10 times, and the changes of the cover images can be more intuitively displayed.

As can be seen from Figure 7, the embedding of the secret images do not produce major modification to the cover images in the proposed scheme, and the human eye cannot recognize the difference between the stego images and the cover images. It can be seen from Figure 7(d)-(f) that the secret image information in Figure 7(c) is not presented in the residual images, and the intensity of the elements in the residual images remain weak even when the residual images are enlarged by 5 times and 10 times. This is because the embedded secret images were scrambled, and the semantic information of the secret images was not leaked with a high probability, indicating that the difference between the cover images and the stego images was difficult to visually perceive.

4.1.3 Anti-Steganalysis of the Model

In order to prove that the proposed scheme had good security and can effectively resist steganalysis tool, by introducing a CNN-based model as a binary classifier to discriminate the cover images and the generated stego images on the tiny dataset, it can be seen that the accuracy of the steganalysis tool was constantly decreasing, indicating that the quality of the stego images generated by the proposed scheme was getting higher and higher during the continuous iterations. To verify the applicability of the proposed scheme on other datasets, the classifier was also used on ImageNet and COCO datasets, and the results were shown in Table 4. Because the parameters in the model network layer of the proposed scheme required training time, certain training time was needed. With the increase of training times, it can be seen that the anti-steganalysis performance of the proposed scheme was gradually improving on different image datasets. The security of the secret images was ensured.

Table 4: The anti-steganalysis performance

Dataset	600	1000	1400
LFW	0.7612	0.7312	0.7189
ImageNet	0.7367	0.7112	0.7034
COCO	0.7339	0.7190	0.7003

In addition, the proposed scheme also used StegExpose [3] steganalysis detection tool to analyze the experimental results, which was a steganalysis tool specially designed to detect least significant bit (LSB) steganography. Using the steganalysis tool to test the proposed model and the model based on AdvSGAN [18], the receiver operating characteristic curves (ROC) of the two models were obtained as shown in Figure 8.

The figure shows that the area under the ROC of the proposed model was close to 0.5, it meant the ROC curve was closer to a random distribution, indicating that the proposed scheme can effectively resist the steganalysis methods that appear in StegExpose.

4.2 Stego Image Quality Analysis

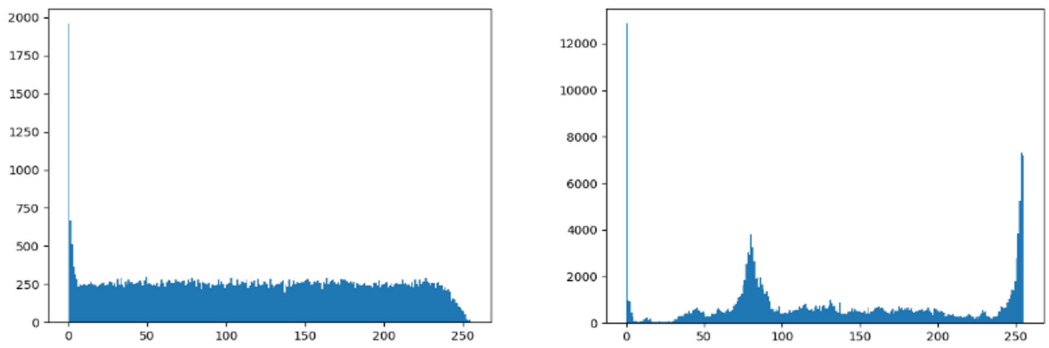
The structural similarity (SSIM) and peak signal to noise ratio (PSNR) are used to evaluate the loss between the cover image and the stego image. The SSIM is a metric based on the HVS and is used to quantify the difference in structural information between two images. The closer the SSIM value is to 1, the closer the image is to human visual perception, and the SSIM calculation method is shown in Equation (13).

$$SSIM(I, I_a) = \frac{(2\varepsilon_I \varepsilon_{I_a} + c_1)(2cov + c_2)}{(\varepsilon_I^2 + \varepsilon_{I_a}^2 + c_1)(\sigma_I^2 + \sigma_{I_a}^2 + c_2)} \quad (13)$$

where I and I_a represent the two images, ε_I and ε_{I_a} represent the mean of I and I_a respectively, σ_I^2 and $\sigma_{I_a}^2$ are the variances of I and I_a. cov is the covariance of I and I_a. c_1 and c_2 are variables to avoid the denominator being zero.

The PSNR is often used to measure the difference between the modified image and the original image, the higher the PSNR value, the closer stego image is to the original cover image. The calculation method of PSNR is shown in Equation (14).

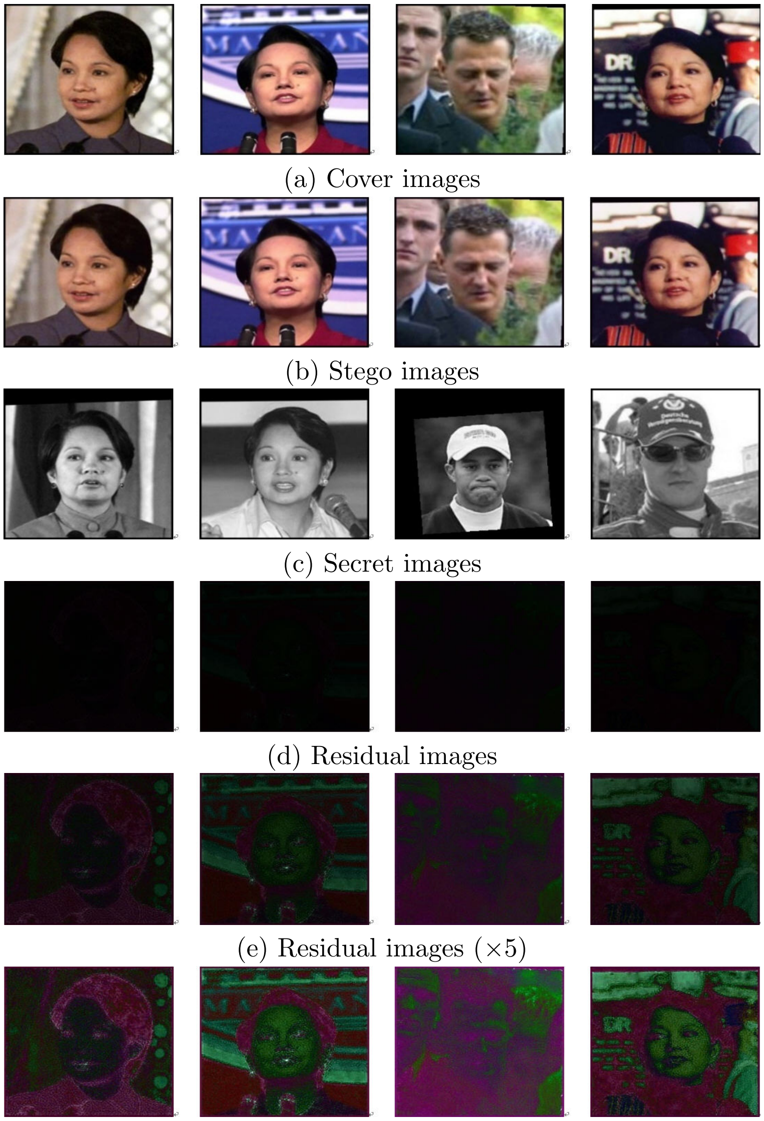
$$PSNR = 10 \times \log_{10} \frac{p_{max}^2}{MSE} \quad (14)$$



(a) Ours scheme

(b) Yang's scheme

Figure 6: Encrypted image histogram analysis



(a) Cover images

(b) Stego images

(c) Secret images

(d) Residual images

(e) Residual images (×5)

(f) Residual images (×10)

Figure 7: Comparison of residual image visualization results

where p_{max} is the maximum value taken for the image pixel value, MSE is the mean square error, and MSE is specifically defined as shown in Equation (9).

The proposed scheme calculated the mean values of PSNR and SSIM for 500 pairs of cover images and stego images on three different datasets, and introduced PSNR-HVS and PSNR-HVSm as evaluation metrics. PSNR-HVS is an improved PSNR metric based on the charac-

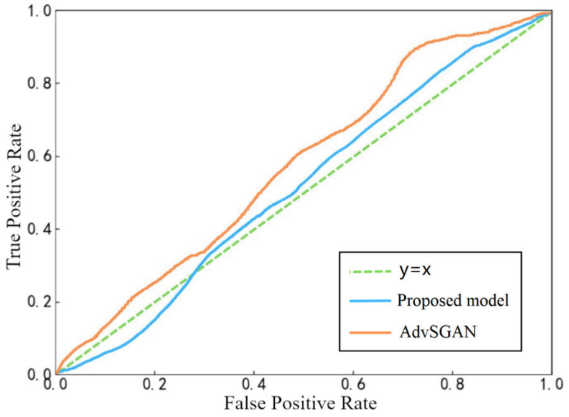


Figure 8: The ROC curves

teristics of HVS, which takes into account the error sensitivity, structural distortion and edge distortion of the images. Table 5 shows the comparison results of cover image loss for different datasets.

Table 5: Cover image loss under different datasets

Dataset	LFW	ImageNet	COCO
SSIM	0.9881	0.9761	0.9737
PSNR	37.9540	36.5502	36.9177
PSNR-HVS	36.8634	35.9805	35.6574
PSNR-HVSm	41.2439	39.6890	40.0456

As can be seen from Table 5, the proposed scheme can maintain good PSNR and SSIM value in the three datasets, the PSNR-HVS value and PSNR-HVSm value also maintained a high level. The loss of transformation from cover image to stego image was less, and better results were obtained under different datasets. The experiments proved that the proposed scheme had less impact on the statistical features of the image while performing information embedding. Table 6 shows the comparison results of PSNR and SSIM values under different steganography schemes.

Table 6: Comparison of PSNR and SSIM values

Schemes		Cover-Stego image	Secret-extracted secret image
Ref. [31]	PSNR SSIM	36.3700 0.9500	35.8400 0.9411
Ref. [22]	PSNR SSIM	**39.7708** 0.9828	43.3571 **0.9626**
Ref. [20]	PSNR SSIM	26.0245 0.7850	24.1432 0.8027
Ref. [25]	PSNR SSIM	36.7800 0.8900	35.9800 0.8100
Ref. [34]	PSNR SSIM	34.6300 0.9573	33.6300 0.9429
Ref. [32]	PSNR SSIM	**42.9721** 0.9710	41.3000 0.9431
Proposed	PSNR SSIM	37.9537 0.9830	36.0326 0.9472

It can be seen from Table 6 that the PSNR values of the proposed scheme stego image and cover image, extracted image and secret image were all greater than 30dB, which indicated that both the stego image and extracted secret image had better visual effects. From the comparison results, the SSIM value between the extracted secret image and the secret image of the proposed scheme was lower than that of the Liu's scheme [22]. The two SSIM values generated by the proposed scheme had great advantages, which were closely related to the network structure and loss function. Both Lin's scheme [20] and Zhang's scheme [34], they proposed scheme embed the grayscale image into the Y channel of the color cover image. From the experimental results, the PSNR and SSIM values of the stego image and the cover image, and the extracted image and the secret image of the proposed scheme were both larger than Lin's scheme [20] and Zhang's scheme [34], this was because the discriminator structure of the proposed scheme was a better aid to the network structure and the loss function provided a good balance between the encoding and decoding tasks. In addition, as the proposed scheme firstly encrypted the secret image before embedding it into the cover image, the secret image entering the network was encrypted, which invariably added noise, which made the network training better and the visual effect of the generated stego image was more deceptive to the human visual system. Compared with the Qin's scheme [25], the experimental results of the proposed scheme also had obvious advantages. Except for the Yang's scheme [31], the Liu's scheme [22] and Yuan's scheme [32] both were the secret images directly embedded into the cover image by the encoding network. The proposed scheme introduced pixel scrambling, and the PSNR value is based on the error between the pixels, the encrypted secret image when the image was recovered through the extraction network, the secret image needed to be further decrypted, which resulted in a lower PSNR value of the proposed scheme than the Liu's scheme [22] and Yuan's scheme [32]. However, compared with the Yang's scheme [31] that also encrypted the secret image and then embedded it into the cover image, the quality of the proposed scheme had been significantly improved in extracting the secret image.

4.3 Robustness Analysis

For common attack translation, rotation, salt and pepper noise, and Gaussian noise, the robustness of the proposed scheme was tested. The stego images were translated horizontally by 6 pixels, vertically by 6 pixels, and horizontally and vertically by 3 pixels simultaneously, the standard deviation of salt and pepper noise was 0.002, and the Gaussian filter used an attack with a template size of 7×7 for the attacks. The trained decoding network was used to extract the secret images, and the SSIM and PSNR values were used to evaluate the similarity between the extracted secret images and the original secret images after the attacks, as shown in Table 7.

As can be seen from Table 7, the proposed scheme can resist some of the image attacks well, and the image extracted from the stego image after the attacks on the secret-containing image, which was still a recognizable image. In these attack experiments, there is less loss of the secret image after the noise attack and Gaussian attack, and the above experiments are common attacks in the image transmission process. Therefore, the proposed scheme had the ability to handle image attacks with some robustness.

4.4 Steganography Capacity

The proposed scheme was to hide the grayscale image in the color image, which had a certain improvement compared with the traditional spatial domain scheme. Effective capacity (EC) was used to represent the embedding rate of secret data, which can reflect the embedding degree of data. The proposed scheme not only took into account the better concealment of stego images, but also had a higher embedding capacity. The size of the embedded secret image and the cover image were both 256 × 256, and the embedded capacity was 8bpp (bit per pixel). The calculation method of the effective capacity is shown in Equation (15).

$$EC = \frac{NS}{NC} \tag{15}$$

where NS was the size of the secret data (in bits) and NC was the number of pixels extracted from the cover image.

Table 8 shows the comparison results of EC between the proposed scheme and the existing traditional steganography schemes [1, 9] and neural network-based schemes [20, 22, 31, 34].

As can be seen from Table 8, compared with traditional data hiding schemes [1, 9], the proposed scheme had a larger effective embedding capacity. As the Yang's scheme [31] was hiding color images in color images, the embedding capacity was 3 times higher than the proposed scheme. Compared with similar schemes [20, 22, 34] that hided grayscale images in color images, each pixel of the cover image of the proposed steganographic model can also hide 8bit data, which achieved the theoretical maximum hiding ability of using a neural network to hide a grayscale image in a color image scheme.

5 Conclusions

In this paper, the high quality image steganography model based on encoder-decoder networks and 2D logistic chaotic encryption was proposed. By encrypting the secret image before embedding, the security of the secret image is greatly improved. At the same time, in order to improve the quality of the stego image, the color cover image is processed by channel separation technology, and the discriminator part is improved so that it can play a better role in adversarial processing, so that the color of the stego image is not distorted and it can more natural. The experimental results show that the steganographic scheme of the proposed model not only has a high embedding capacity, but also the generated stego image has strong imperceptibility and robustness, which ensures the security and anti-attack of the secret image. In addition, good results have been obtained on the 3 classic datasets of LFW, ImageNet, and COCO. However, the model proposed in this paper still has shortcomings in extracting image quality, and the model fails to solve the information hiding problem of color secret images. In the future, we will continue to study the relationship between the encoder and the decoder, and make the entire network architecture be more balance.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61862041).

References

[1] A. AbdelRaouf, "A new data hiding approach for image steganography based on visual color sensitivity," *Multimedia Tools and Applications*, vol. 80, no. 15, pp. 23393–23417, 2021.

[2] M. Alkhelaiwi, W. Boulila, J. Ahmad, *et al.*, "An efficient approach based on privacy-preserving deep learning for satellite image classification," *Remote Sensing*, vol. 13, no. 11, pp. 2221–2247, 2021.

[3] B. Boehm, "Stegexpose-a tool for detecting LSB steganography," *arXiv preprint arXiv:1410.6656*, 2014.

[4] Y. T. Chen, H. P. Zhang, L. W. Liu, *et al.*, "Research on image inpainting algorithm of improved total variation minimization method," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–10, 2021.

[5] M. Dalal and M. Juneja, "Steganography and steganalysis (in digital forensics): a cybersecurity guide," *Multimedia Tools and Applications*, vol. 80, no. 4, pp. 5723–5771, 2021.

[6] J. Deng, W. Dong, R. Socher, *et al.*, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Florida, USA, Jun 2009.

[7] Y. Ding, G. Z. Wu, D. J. Chen, *et al.*, "DeepEDN: a deep-learning-based image encryption and decryption network for internet of medical things," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1504–1518, 2020.

Table 7: Anti-attack experiment pictures and data

Attacks	Example images				PSNR	SSIM
	Cover image	Attacked Stego image	Secret image	Extracted image		
Translate 6 pixels horizontally					28.97	0.4537
Translate 6 pixels vertically					26.38	0.4289
Translate 3 pixels horizontally and vertically					30.16	0.5697
Rotate 5 degrees					29.61	0.6238
Salt and pepper noise					30.72	0.8762
Gaussian filtering					32.19	0.8549

Table 8: Comparison of effective capacity of different models

Schemes	Secret image size(bits)	Cover image size (bits)	EC (bpp)
Ref. [1]	$< 128 \times 128 \times 8$	521×512	< 3
Ref. [9]	$\leq 256 \times 256 \times 8$	256×256	< 2
Ref. [31]	256×256	256×256	24
Ref. [22]	256×256	256×256	8
Ref. [20]	256×256	256×256	8
Ref. [34]	256×256	256×256	8
Proposed	256×256	256×256	8

[8] X. T. Duan, L. Li, Y. Su, *et al.*, "DIGDH: A novel framework of difference image grafting deep hiding for image data hiding," *Symmetry*, vol. 14, no. 151, pp. 1–12, 2022.

[9] R. A. El-Shahed, M. ElBery, H. Moushier, *et al.*, "Image hiding using upper-lower decomposition technique," *International Journal of Intelligent Computing and Information Sciences*, vol. 21, no. 1, pp. 95–103, 2021.

[10] Z. J. Fu, F. Wang, and X. Cheng, "The secure steganography for hiding images via gan," *EURASIP Journal on Image and Video Processing*, vol. 2020, no. 1, pp. 1–18, 2020.

[11] Z. J. Fu, F. Wang, X. M. Sun, *et al.*, "Research on steganography of digital images based on deep learning," *Chinese Journal of Computers*, vol. 43, no. 9, pp. 1656–1672, 2020.

[12] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pp. 2672–2680, Massachusetts, MA, USA, Dec 2014.

[13] J. Hayes and G. Danezis, "Generating steganographic images via adversarial training," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 1951–1960, California , USA, Dec 2017.

[14] I. Kich, E. B. Ameur, Y. Taouil, *et al.*, "Image steganography scheme using dilated convolutional network," in *The 12th International Conference on*

Information and Communication Systems (ICICS), pp. 305–309, Valencia, Spain, May 2021.

[15] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, *et al.*, "Labeled faces in the wild: A survey," *Springer*, pp. 189–248, 2016.

[16] E. L. Li, Z. J. Fu, and J. F. Chen, "Adaptive steganography based on image edge enhancement and automatic distortion learning," in *The 11th International Conference on Image and Graphics*, pp. 155–167, Cham, Switzerland, Dec 2021.

[17] F. Y. Li, Z. L. Yu, and C. Qin, "GAN-based spatial image steganography with cross feedback mechanism," *Signal Processing*, vol. 190, pp. 108341–108353, 2022.

[18] L. Li, M. Y. Fan, and D. F. Liu, "AdvSGAN: Adversarial image steganography with adversarial networks," *Multimedia Tools and Applications*, vol. 80, no. 17, pp. 25539–25555, 2021.

[19] T. Y. Lin, M. Maire, S. Belongie, *et al.*, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Cham, Switzerland, Sept 2014.

[20] W. J. Lin, X. K. Zhu, W. J. Ye, *et al.*, "An improved image steganography framework based on Y channel information for neural style transfer," *Security and Communication Networks*, vol. 2641615, pp. 1–12, 2022.

[21] J. Liu, T. P. Zhou, Z. Zhang, *et al.*, "Digital cardan grille: a modern approach for information hiding," in *Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence*, pp. 441–446, Shenzhen, China, Dec 2018.

[22] L. S. Liu, L. Z. Meng, Y. J. Peng, *et al.*, "A data hiding scheme based on U-Net and wavelet transform," *Knowledge-Based Systems*, vol. 223, pp. 107022–107031, 2021.

[23] L. Mancy and S. M. C. Vigila, "A new diffusion and substitution-based cryptosystem for securing medical image applications," *International Journal of Network Security*, vol. 22, no. 5, pp. 793–800, 2020.

[24] P. Pan, Z. M. Wu, C. Yang, *et al.*, "Double-matrix decomposition image steganography scheme based on wavelet transform with multi-region coverage," *Entropy*, vol. 24, no. 2, pp. 246–257, 2022.

[25] J. H. Qin, J. Wang, Y. Tan, *et al.*, "Coverless image steganography based on generative adversarial network," *Mathematics*, vol. 8, no. 9, pp. 1394–1405, 2020.

[26] K. Sharma, A. Aggarwal, T. Singhania, *et al.*, "Hiding data in images using cryptography and deep neural network," *Journal of Artificial Intelligence and Systems*, vol. 1, no. 1, pp. 143–162, 2019.

[27] W. Z. Shi and S. H. Liu, "Hiding message using a cycle generative adversarial network," *ACM Transactions on Multimedia Computing Communications, and Applications (TOMM)*, 2022.

[28] D. Volkhonskiy, I. Nazarov, and E. Burnaev, "Steganographic generative adversarial networks," in *Twelfth International Conference on Machine Vision*, p. 11433M, Amsterdam, Switzerland, Nov 2019.

[29] G. S. Xu, H. Z. Wu, and Y. Q. Shi, "Structural design of convolutional neural networks for steganalysis," *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 708–712, 2016.

[30] J. H. Yang, D. Y. Ruan, J. W. Huang, *et al.*, "An embedding cost learning framework using gan," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 839–851, 2020.

[31] X. Y. Yang, X. L. Bi, J. Liu, *et al.*, "High-capacity image steganography algorithm combining image encryption and deep learning," *Journal on Communications*, vol. 42, no. 9, pp. 96–105, 2021.

[32] C. Yuan, H. X. Wang, P. S. He, *et al.*, "GAN-based image steganography for enhancing security via adversarial attack and pixel-wise deep fusion," *Multimedia Tools and Applications*, vol. 81, no. 5, pp. 6681–6701, 2022.

[33] K. A. Zhang, A. Cuesta-Infante, L. Xu, *et al.*, "SteganoGAN: High capacity image steganography with GANs," *arXiv preprint arXiv:1901.03892*, 2019.

[34] R. Zhang, S. Q. Dong, and J. Y. Liu, "Invisible steganography via generative adversarial networks," *Multimedia tools and applications*, vol. 78, no. 7, pp. 8559–8575, 2019.

[35] J. R. Zhu, R. Kaplan, J. Johnson, *et al.*, "Hidden: Hiding data with deep networks," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 657–672, Munich, Germany, Sept 2018.

[36] L. Q. Zhu, Y. Guo, L. Q. Mo, *et al.*, "DGANS: robustness image steganography model based on double gan," *Journal on Communications*, vol. 41, no. 1, pp. 125–133, 2020.

Biography

Qiu-yu Zhang Researcher/Ph.D. supervisor, graduated from Gansu University of Technology in 1986, and then worked at school of computer and communication in Lanzhou University of Technology. He is vice dean of Gansu manufacturing information engineering research center, a CCF senior member, a member of IEEE and ACM. His research interests include network and information security, information hiding and steganalysis, multimedia communication technology.

Xue-wen Hu is currently a master student of the School of Computer and Communication, Lanzhou University of Technology, China. She received the BS degrees in software engineering from Lanzhou University of Technology, Gansu, China, in 2019. Her research interests include network and information security, image information hiding and steganography, multimedia data security.

Zhen Wang is currently a master student of the School of Computer and Communication, Lanzhou University of Technology, China. He received the BS degrees in software engineering from Heilongjiang University of Science and Technology, Heilongjiang, China, in 2018. His research interests include network and information security, multimedia data security.

A Deep Learning Algorithm for Detecting Bot Infections in Host Networks

Tianru Hu[1], Xiuli Lu[2], and Hongyan Zuo[1]
(Corresponding author: Xiuli Lu)

Department of Information Engineering and Management, Baoding Technical College of Electric Power[1]
School of Information Science and Engineering, Baoding University of Technology[2]
Baoding, Hebei 071000, China
Email: blx566319@163.com

(Received Mar. 21, 2022; Revised and Accepted Mar. 23, 2023; First Online Apr. 13, 2023)

Abstract

Botnets have become one of the main problems of current Internet security risks. Attackers can launch attacks based on botnets to induce online fraud, network information leakage, and other behaviors. Traditional botnet detection methods are not flexible, so this paper used the convolutional neural network (CNN), a deep learning algorithm, to detect bot program-infected host networks. Through data collection and pre-processing, a network model was built to perform feature learning of domain name bytes to detect whether the host network is infected with a botnet. The effectiveness of the CNN model in detecting botnets was verified by comparing it with the results of random forest and support vector machine. It was found that the CNN model had a precision of 97.14%, a recall rate of 97.43%, an F1 value of 97.28%, and an average recognition speed of 2.05 s, all of which were higher than those of both the random forest and support vector machine approaches. The results prove that the CNN model possesses high accuracy for detecting botnets and can be used for detecting bot programs in host networks.

Keywords: Deep Learning Algorithm; Botnet; Convolutional Neural Network

1 Introduction

With the rapid development of the Internet, people's work, life and study are gradually related to the Internet, such as working via DingTalk, online shopping, social communication with software, and online learning. However, more and more security risks appear, such as spam and information leakage. One of the most common risk is host network infection by bot program. A botnet is a set composing of many infected hosts; after receiving the command released by the attacker, the hosts will execute various scams or cyber attacks [1]. An attacker can take control of an infected host with malware and use Distributed Denial of Service (DDoS) attacks to prevent legitimate users from logging in to access the Internet [4, 7, 16].

The traditional detection method used for botnet detection is to set threshold conditions for network traffic feature indicators, and when the threshold conditions are not met then it is considered botnet traffic. However, this approach requires a high level of domain knowledge and is not flexible enough to detect unknown botnet traffic in a timely manner. Nguyen *et al.* proposed a collaborative machine learning model to detect Internet of Things (IoT)-botnets, which had an accuracy of 99.37% for detecting a dataset containing 5023 IoT-botnets and 3888 benign samples [13]. Koroniotis *et al.* discussed the feature extraction process based on some benign and botnet scenarios through a new dataset-Bot-IoT dataset and proved the reliability of the dataset using several statistical and machine learning methods [8]. Venturi *et al.* presented the dataset in CSV file format by using deep reinforcement learning (DRL) techniques. They made small modifications to the initial malicious samples to achieve misclassification and proposed new techniques for effective defense against botnets [17].

Garre *et al.* proposed that the initial stages of botnet infection could be detected by machine learning (ML) techniques. They found ML techniques could detect new SSH infections by designing an SSH-based high interaction honeypots approach to generate a dataset consisting of executed commands and network information [12]. Idrissi *et al.* found that compared to other deep learning techniques (e.g., simple recurrent neural network, long short-term memory, and gated recurrent unit), the deep learning-based intrusion detection system obtained better results, with a validation correct rate of 99.94%, a validation loss of 0.58%, and prediction execution time less than 0.34 ms [5]. This paper detected whether the host network was infected with a bot program by building a convolutional neural network (CNN) model. After the initial data collection, data preprocessing was performed to convert the data into 32*32 grayscale images, and the images were cut into uniform length. Then, a CNN model was established and optimize by continuously adjusting the parameters. Finally, the optimal model was used to detect whether the host network was infected with a botnet. This work provides a theoretical basis for the detection of bot program infection in host networks with a CNN.

2 Bot Program Detection Method

2.1 Convolutional Neural Network Model

A CNN [2] is one of the most typical algorithms in deep learning algorithms and have been used many times in the field of network security detection. The following is a brief description for the CNN model established in this paper [15].

1) Input layer: Every feature of the network is treated as a pixel in an image, and the network domain name is transformed into an image input.

2) The first convolutional layer: This layer uses 32 5*5 convolutional kernels with a step size of 1. The RELU function [10] is used as the network activation function for nonlinear transformation, and 32 28*28 feature maps are output. The function only determines whether the input function value x is within the positive interval, which is less computational. The mathematical formula of the RELU function is:

$$Relu(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases}$$

3) The first pooling layer: This layer uses a 2*2 filter and outputs 32 14*14 feature maps. Its specific calculation formula is:

$$y_{mn} = f(w\frac{1}{S_1 S_2}\sum_{j=0}^{S_2-1}\sum_{i=0}^{S_1} x_{m*S_{1+i},n*S_{2+j}} + b),$$

where $x_{m*S_{1+i},n*S_{2+j}}$ denotes the pixel value of point $(m*S_{1+i}, n*S_{2+j})$ and y_{mn} denotes the output value after the pooling operation.

4) The second convolutional layer: This layer uses 64 3*3 convolutional kernels with a step size of 1. The RELU function is used as the network activation function, and 64 12*12 feature maps are output.

5) The second pooling layer: It has the same specification as the previous pooling layer; this layer uses a 2*2 filter and finally outputs 64 6*6 feature maps.

6) Fully connected layer: It is mainly responsible for connecting the feature maps extracted from the previous layer and classifying the input images based on these features based on the training data. This layer consists of 1024 neurons, and the output is a 1024-dimensional vector.

7) Output layer: The output of this layer is the final classification result of the model. The classification results of the proposed model are normal network and botnet; thus, the number of neurons of the output layer is 2.

2.2 Accelerated Training Model Method

The regularization method "Dropout" [3] is used in the training model and is one of the ways to prevent the network from fitting. Its main working principle is that some neurons are eliminated in the training process of the deep learning network and the remaining neurons remain unchanged and still participate in the training of the network. The neurons removed in that process are chosen randomly and the values retained before the removal will be restored in the next time of network training. This way reduces the correlation between neurons and prevents a local feature from fitting the network. To prevent network fitting, the Dropout parameter is set as 0.2 in this paper.

2.3 Loss Function Selection

As the characters and other aspects of botnet domain names are different, there is a botnet domain name data imbalance in the overall dataset. In this paper, we address this situation at the algorithmic level by using a loss function to relieve the data imbalance phenomenon. The loss function is used to describe the difference between the predicted and actual values when the model is trained, represented by C. When the value of C is large, it means that the network model detection is poor; when the value of C is small, it means that the network model detection is good; when C = 0, the network model detection is the best. The loss function used in this study is the cross-entropy loss function [11], and its specific formula is:

$$C = -\sum_{n}(p_n * \log q_n),$$

where p denotes the desired output and q denotes the actual output of the network detection.

3 Experimental Analysis

3.1 Data Collection and Processing

The dataset used in this paper mainly contained normal network domain names and botnet domain names. The botnet domain names came from the DGA malicious domain names in 360 network security lab [14], while the normal network domain names were the normal and legitimate domain names in alexa. The data processing operations are as follows.

First of all, the dataset was in pcap format, so the pcap file data were read first, the network domain names were intercepted according to the uniform length, and 0*00 was supplemented at the end of those shorter than the uniform length, and the uniform length was set as 1,024 bytes. Since the above mentioned training model adopted the regularization method "Dropout", data normalization was not needed.

Secondly, in order to ensure that the extracted features had the same dimension, the network traffic was transformed into 32*32 images before convolution, and the images were processed to be gray. The data set was divided according to the ratio of 4:6. 40% of the data were classified as the training samples to train the CNN model to learn the features of domain name bytes and continuously optimize the model; 60% of the data were used as the test samples for the optimized CNN model to detect botnet to judge the feasibility of the model.

3.2 Experimental Design

The overall experimental design had four steps. The first step was data collection and processing. A sufficient number of experimental network domain names was needed and used as the initial data set. The collected data network were converted into images, followed by gray processing. Then, the network domain names were intercepted into the same length to achieve the standard.

The second step was the training of the CNN model. In order to optimize the bot program detection network model, it was trained to learn the features of domain names. The processed images were divided into two parts according to the ratio of 4:6. 40% of the images were used as training samples, and the data set was input into the network model to constantly adjust the parameters to achieve the optimal effect.

The third step was the testing of the CNN model. After obtaining the optimal network model through the above steps, the remaining 60% of the images were input the network model as the test set to detect whether the host network was infected with a bot program. The last step was to compare the experimental results of the CNN model with random forest [9] and support vector machine [6] to verify whether the CNN model can detect the host infected with bot program. In order to ensure the authenticity and reliability of the data in this experiment, the initial parameters were set consistently. The details are shown in Table 1.

Table 1: Initial parameters

Parameter Category	Setting
Optimizer	Adam
Learning rate	0.01
Number of iterations	25
Batch size	150

3.3 Evaluation Indicators

3.3.1 Secondary Evaluation Indexes of Confusion Matrix

The precision, recall rate, and F_1 value among the secondary evaluation indexes of the confusion matrix were selected. The precision represented the ratio of the number of botnets correctly detected by the model to the total number of botnets detected by the model, and its calculation formula is:

$$P = \frac{TP}{TP + FP}$$

The recall rate represents the ratio of the number of botnets correctly detected by the model to the total number of botnets in the test set. It is calculated by the following equation:

$$R = \frac{TP}{TP + FN}$$

The meanings of the parameters mentioned in the above two equations are specified in Table 2.

Table 2: Meanings of equation parameters

	Forecast (positive)	Forecast (negative)
Actual (true)	TP	TN
Actual (false)	FP	FN

The F_1 value is a combination of both precision and recall rate. The larger the F_1 value, the better the network model detection effect, and vice versa. Its calculation formula is:

$$F_1 = \frac{precision * recallrate}{precision + recallrate} * 2.$$

3.3.2 Receiver Operator Characteristic Curve

The receiver operator characteristic (ROC) curve graph obtained used the false alarm rate as the horizontal coordinate and the recall rate as the vertical coordinate. When the ROC curve in the picture was far away from the line y=x, it indicated that the detection effect of the model was good. The ROC curve is often used in combination with area under the curve (AUC), which is the area enclosed with the coordinate axis under the ROC curve, and its value ranges from 0.5 to 1. When the value of the AUC area was closer to 1, it indicated that the detection effect of the network model was good.

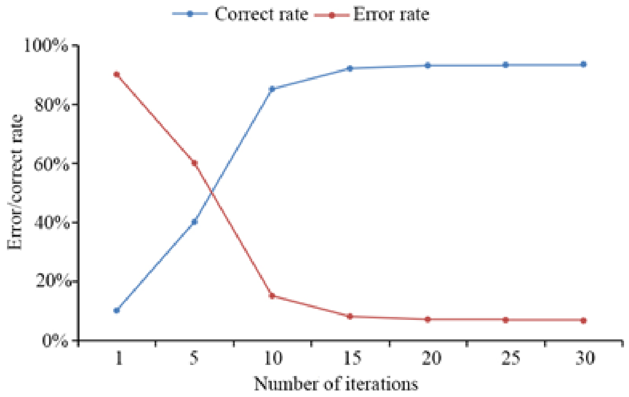


Figure 1: Relationships between error rate, correct rate, and number of iterations

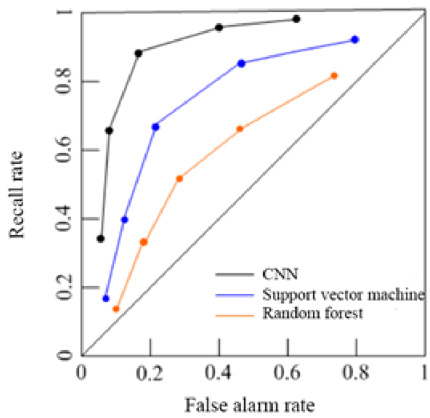


Figure 2: ROC curves and AUC of different detection models

3.4 Analysis of Results

If the total number of iterations decreased, the network model would learn domain name features for fewer times, which was likely lead to feature missing in the training process. It was seen from the line graph in Figure 1 that the error rate decreased and the correct rate increased with the increase of the number of iterations. However, after a certain number of iterations, the variations of the error rate and the correct rate of the model decreased until they were negligible, despite the increasing number of iterations. Thus, it was concluded that the error rate and correct rate did not increase all the time with the increase of the number of iterations. Therefore, the total number of iterations of the network model was determined as 25 for the practical consideration.

The ROC curves of the three detection models and the corresponding AUC are clearly seen in Figure 2. The prediction result of the CNN model was the farthest from the line Y=X, and its UC was the closest to 1. The prediction result of the random forest model was the closest to the line Y=X, and its AUC was the smallest. The three models were ranked in order of distance and area, and it was seen that the CNN model had the best effect and high accuracy in detecting botnets, so it can be used to detect hosts infected with bot programs.

It was observed in Table 3 that the precision of the random forest model to detect botnet was 85.24%, the precision of the support vector machine model was 90.21%, and the precision of the CNN model was 97.14%; in terms of the recall rate, the CNN model was 11.96% higher than the random forest model and 7.05% higher than the support vector machine model; in terms of the F_1 value, the CNN model was 11.93% larger than the random forest model and 6.99% larger than the support vector machine model. In terms of the average recognition speed, the CNN model was faster than the support vector machine and random forest models. Therefore, it was concluded that the effects of the three models in detecting host network infected with bot programs were the CNN model, the support vector machine model, and the random forest model in descending order, and the CNN model was much better than both random forest and support vector machine models in detecting bot infection in host networks.

4 Conclusion

This paper briefly introduced botnet and CNN model and detected whether the host network was infected with a bot program by the CNN model. After the initial data collection, data preprocessing was performed to convert the network domain names into 32*32 grayscale images and to cut the length of the network domain names to a uniform length. Then, a CNN model was established to learn the features of the domain name bytes, and the parameters were constantly adjusted to optimize the model. Finally, the optimal model was used to detect whether the host network was infected with a bot program. The research results showed that the precision of the CNN model to detect botnet was 97.14%, the recall rate was 97.43%, the F_1 value was 97.28%, and the average recognition speed was 2.05 seconds. The CNN model was much higher than both random forest and support vector machine models in terms of all the evaluation indicators. It indicates that the CNN model possesses high accuracy for botnet detection and can be used for detecting host networks infected with bot programs.

References

[1] Y. Chen, B. Pang, G. Shao, G. Wen, X. Chen, “Erratum to ’DGA-based botnet detection toward imbalanced multiclass learning’,” *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 790-790, 2021.

[2] M. V. O. De Assis, L. F. Carvalho, J. Rodrigues, J. Lloret, M. L. Proenca Jr, “Near real-time security system applied to SDN environments in IoT networks

Table 3: Comparison of detection results between different ways

Methods	Average recognition speed	Precision	Recall rate	F_1 value
Random forest	3.26 (S)	85.24%	85.47%	85.35%
Support vector machine	2.74 (S)	90.21%	90.38%	90.29%
CNN	2.05 (S)	97.14%	97.43%	97.28%

using convolutional neural network," *Computers & Electrical Engineering*, vol. 86, no. 3, pp. 1-16, 2020.

[3] J. Hu, Y. Chen, L. Zhang, Z. Yi, "Surrogate dropout: Learning optimal drop rate through proxy," *Knowledge-Based Systems*, vol. 206, pp. 1-9, 2020.

[4] M. S. Hwang, S. K. Chong, , H. H. Ou, "The moderately hard DoS-resistant authentication protocol on client puzzles", *Informatica*, vol. 27, pp. 31-48, 2016.

[5] I. Idrissi, M. Boukabous, M. Azizi, O. Moussaoui, H. E. Fadili, "Toward a deep learning-based intrusion detection system for IoT against botnet attacks," *IAES International Journal of Artificial Intelligence*, vol. 10, no. 1, pp. 110-120, 2021.

[6] S. Jagadeesan, B. Amutha, "An efficient botnet detection with the enhanced support vector neural network," *Measurement*, vol. 176, pp. 1-10, 2021.

[7] J. Jeeshitha, G. Ramakoteswararao, "Extensive study on DDoSBotNet attacks in multiple environments using deep learning and machine learning techniques," *ECS Transactions*, vol. 107, no. 1, pp. 15181-15193, 2022.

[8] N. Koroniotis, N. Moustafa, E. Sitnikova, B. P. Turnbull, "Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-IoT dataset," *Future Generation Computer Systems*, vol. 100, pp. 779-796, 2019.

[9] V. Kothandapani, "Big data analytics framework for peer-to-peer botnet detection using random forest and deep learning," *International Journal of Computer Science and Information Security*, vol. 15, no. 11, pp. 269-277, 2019.

[10] N. Kulathunga, N. R. Ranasinghe, D. Vrinceanu, Z. Kinsman, L. Huang, Y. Wang, "Effects of non1inearity and network architecture on the performance of supervised neura1 networks," *Algorithms*, vol. 14, no. 2, pp. 1-17, 2021.

[11] X. Li, L. Yu, D. Chang, Z. Ma, J. Cao, "Dual cross-entropy loss for small-sample fine-grained vehicle classification," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4204-4212, 2019.

[12] J. T. Martínez Garre, M. Gil Pérez, A. Ruiz-Martínez, "A novel machine learning-based approach for the detection of SSH botnet infection," *Future Generation Computer Systems*, vol. 115, pp. 387-396, 2021.

[13] G. L. Nguyen, B. Dumba, Q. D. Ngo, H. V. Le, T. N. Nguyen, "A collaborative approach to early detection of IoT Botnet," *Computers and Electrical Engineering*, vol. 97, pp. 1-13, 2022.

[14] V. C. Nguyen, N. T. Doan, A. T. Tong, V. L. Hoang, H. S. Le, T. K. S. Nguyen, "A new method to classify malicious domain name using Neutrosophic sets in DGA botnet detection," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, no. 4 Pt.2, pp. 1-14, 2020.

[15] M. Panda, A. Mousa, A. E. Hassanien, "Developing an efficient feature engineering and machine learning model for detecting IoT-Botnet cyber attacks," *IEEE Access*, vol. 9, pp. 91038-91052, 2021.

[16] J. R. Sun, M. S. Hwang, "A new investigation approach for tracing source IP in DDoS attack from proxy server", in *Intelligent Systems and Applications*, pp. 850-857, 2015.

[17] A. Venturi, G. Apruzzese, M. Andreolini, M. Colajanni, M. Marchetti, "DReLAB - Deep REinforcement learning adversarial botnet: A benchmark dataset for adversarial attacks against botnet intrusion detection systems," *Data in Brief*, vol. 34, pp. 106631, 2021.

Biography

Tianru Hu was born in Baoding, Hebei, China, in 1978. From 1997 to 2001, she studied in Yanshan University and received her bachelor's degree in 2001. From 2005 to 2010, she studied in North China Electric Power University and received her Master's degree in 2010. Currently, she works in Baoding Technical College of Electric Power. She has published a total of 19 papers. Her research interest is computer technology.

Xiuli Lu was born in Shandong, China, in 1982. From 2001 to 2005, she studied in Lanzhou University and received her bachelor's degree in 2005. From 2005 to 2008, she studied in Lanzhou University and received her master's degree in 2008. From 2010 to now, she worked in Baoding University of Technology. Her research interests are included program design and development.

Hongyan Zuo was born in Baoding, Hebei, China, in 1973. From 1992 to 1996, she studied in Wuhan University and received her bachelor's degree in 1996. From 2004 to 2008, she studied in North China Electric Power University and received her Master's degree in 2008. Currently, she works in Baoding Technical College of Electric Power. She has published a total of 6 papers. Her research interest is computer technology.

WordDeceiver: Black Box Attack on Chinese Text Classification

Yan Gong[1], Xiao-Lin Zhang[1], Yong-Ping Wang[1], Li-Xin Liu[2], and En-Hui Xu[3]

(Corresponding author: Xiaolin Zhang)

School of Information Engineering, Inner Mongolia University of Science and Technology[1]
Baotou 014010, China
Email: zhangxl6161@163.com
School of Information, Renmin University of China[2]
Beijing 100872, China
China Nanhu Academy of Electronics and Information Technology[3]
Jiaxing 314000, China

(Received Oct. 3, 2022; Revised and Accepted Mar. 3, 2023; First Online Apr. 14, 2023)

Abstract

Aiming at the problem that DNNs-based text classification systems are vulnerable to adversarial example attacks, a black-box adversarial attack method of adversarial example generation for Chinese text classification, WordDeceiver, is proposed. In this method, we use the glyph and phonetic features of Chinese characters to construct adversarial search space, determine the replacement order by word saliency and classification probability, generate adversarial examples using word substitution strategy, and design a new method to improve the semantic similarity between the adversarial examples and the original samples. The effectiveness and transferability are verified on different classification datasets using two mainstream models. The experimental results show that WordDeceiver can preserve the original semantics and grammatical correctness to some extent and can be effectively transferred to other models and cloud platforms.

Keywords: Adversarial Attack; Black Box; Chinese Text Classification; Deep Neural Network

1 Introduction

Deep neural network-based [25] machine learning methods have been widely used in many fields, such as computer vision [9,20], speech recognition [23], natural language processing [3,22], and malware detection [7,8,17]. However, due to the characteristics of local linearity, neural networks face the threat of adversarial attacks, and Szegedy *et al.* [18] first proposed adding imperceptible perturbations to an image in an image classification task could induce misclassification of the model. Since then, research on adversarial attack methods in the field of computer vision has been carried out, for example, the classical attack algorithms are FGSM [5], PGD [12], C&W [1], etc.

Natural language processing, which includes tasks such as sentiment analysis [24], text classification [13], machine translation [16], and question-and-answer systems [10], also suffers from security issues. Unlike images, text has discrete data features, complex syntactic rules, and abstract semantic forms, making it difficult to transfer methods commonly used in computer vision to the field of natural language processing.

In the text domain, Papernot *et al.* [14] used LSTM to generate adversarial examples by replacing the word vectors of randomly selected words in the embedding layer with the nearest neighboring word vectors in the vector space to induce model misclassification; Gao *et al.* [4] proposed an attack algorithm DeepWordBug design a word importance calculation function based on the output of the observed model under black box conditions, find the keywords in the text, and use insertion, replacement, and swapping the positions of the letters before and after, and deletion operations to generate an adversarial example spoofing classifier. The above methods are implemented in English text. Since there are obvious differences between English and Chinese in terms of character types, character lengths, and phonological features, the methods for generating adversarial examples for English text cannot be directly applied to Chinese text.

At present, the research of Chinese text-oriented adversarial attack methods is in the initial stage, Wang *et al.* [21] proposed a character-level black box attack method WordHanding that can attack Chinese emotion classification systems, using homophone substitution to mislead the LSTM, achieving a better attack effect and a small number of perturbed words, but with a single substitution method and insufficient diversity of adversarial examples; Tong *et al.* [19] proposed a method to generate adversarial examples under word-level black box conditions, CWordAttacker, which uses a directed word score deletion mechanism to locate important words and generates adversarial examples through four attack strategies: traditional character replacement, pinyin rewriting, phrase disassembly, and order perturbation, but the perturbation rate is large and fails to maintain a high semantic similarity; Dou *et al.* [6] improved the method of Gao *et al.* to propose FastWordBug, a fast adversarial example generation method, to modify words that are often wrong and construct adversarial examples quickly, but failed to improve the success rate of the attack significantly.

Based on previous research work, a black-box adversarial attack method for Chinese text classification, WordDeceiver, is proposed. The adversarial search space is constructed by analyzing the glyph and phonetic features of Chinese characters, and the replacement order is determined by word saliency and classification probability. The word substitution strategy generates adversarial examples, and a new method is designed to improve the semantic similarity between the adversarial examples and the original samples while ensuring the success of the attack, and effectively realizing the adversarial attack on Chinese text under the multi-scene classification task. We summarize the main contributions of this study as follows:

1) A Chinese text adversarial attack method WordDeceiver under black-box conditions is proposed, which can generate adversarial examples by adding only minor perturbations to the input text without knowing the internal parameters of the target model and is suitable for tasks such as sentiment analysis, spam detection, and news classification.

2) A method is designed to combine Chinese character phonetic and glyph feature to construct the adversarial search space, which effectively improves the grammatical correctness and enhances the imperceptibility of the adversarial examples.

3) The replacement order is determined by word saliency and classification probability and the adversarial examples are generated using the word substitution strategy to reduce the modification rate while improving the attack success rate.

4) A new method is designed that can improve the similarity between the adversarial example and the original sample to enhance the semantic consistency while guaranteeing the success of the attack.

5) Experiments on three publicly available datasets, by attacking CNN [15] and BiLSTM [11] models, resulted in an accuracy reduction of about forty-five percent and a high transferability.

2 Attack Design

2.1 Problem Formulation

For N texts $X = x_1, x_2, ..., x_N$ in the dataset, whose corresponding classification labels are $Y = y_1, y_2, ..., y_M$, F is the classification model that learns the mapping relation $f\colon X \rightarrow Y$ from the input text $x \in X$ to the label $y \in Y$ such that the original text is classified with maximum probability as the correct label y_{true}, as shown in Equation (1):

$$\underset{y_i \in y}{\operatorname{argmax}} P\left(y_i \mid x\right) = y_{\text{true}} \tag{1}$$

Adversarial example x^* is generated by adding a small perturbation Δx to the original input text x, thus forcing the deep learning model F to misclassify, as shown in Equation (2):

$$\underset{y_i \in y}{\operatorname{argmax}} P\left(y_i \mid x^*\right) \neq y_{\text{true}} \tag{2}$$

The definition of the adversarial example x^* is shown in Equation (3):

$$\begin{gathered} x^* = x + \Delta x, \quad \|\Delta x\|_{\mathrm{p}} < \epsilon \\ \underset{y_i \in y}{\operatorname{argmax}} P\left(y_i \mid x\right) \neq \underset{y_i \in y}{\operatorname{argmax}} P\left(y_i \mid x^*\right) \end{gathered} \tag{3}$$

At the same time, the adversarial example after adding the perturbation should satisfy the imperceptibility, so it is achieved by putting constraints on the perturbation, as shown in Equation (4:

$$\|\Delta x\|_p = \left(\sum_{i=1}^{n} \left|w_i^* - w_i\right|^p\right)^{\frac{1}{p}} \tag{4}$$

The perturbation is restricted using p-parameters, which are usually L_0, L_2, and L_∞. The original input text can be represented as $x = w_1 w_2 ... w_i ... w_n$, where $w_i \in D$ is a word and D is a word dictionary. In order to satisfy the semantic consistency, the original text is modified by using the words in the adversarial search space obtained based on phonetic and glyph encoding construction, and the maximum modification threshold is set to constrain the modification magnitude, as shown in Equation (5):

$$F\left(x^*\right) \neq F(x), \operatorname{Cost}\left(x^*, x\right) \leq \sigma \tag{5}$$

where $Cost(\cdot)$ denotes the cumulative frequency of text modification and σ denotes the maximum threshold of modification.

2.2 WordDeceiver

In this section, we focus on four parts: 1) how to construct the adversarial search space based on phonetic and glyph encoding; 2) how to design a word substitution strategy; 3) how to determine the replacement order; and 4) how to optimize the adversarial examples.

2.2.1 Building an Adversarial Search Space Based on Phonetic and Glyph Encoding

The quality of the adversarial search space determines the quality of the adversarial examples. Therefore, the text combines the glyph and phonetic characteristics of Chinese characters to construct the adversarial search space to improve the quality of the adversarial examples. Firstly, the Chinese characters in the dictionary are encoded with their phonetics and glyph, secondly, the Hamming distance between each character and other characters after encoding is calculated, and finally, the similarity is calculated based on the Hamming distance, and finally, the top K characters are selected and added to the adversarial search space in descending order of similarity.

1) Phonetic and glyph encoding methods
The encoding includes both phonetic and glyph parts.

Phonetic part: The phonetic structure of Chinese characters consists of consonants, vowels, and tones, so encoding phonetics should also include these three parts. For some Chinese characters with a vowel between the consonant and the vowel, such as u in nuan, i in miao, etc., an additional coding bit is needed.

The first part represents the consonant, there are 23 consonants, which are represented by five binary digits. To weaken the difference between flat and warble consonants in the later calculation of similarity, the same encoding is used for zh and z, ch and c, and sh and s.

The second part represents the vowel. There are 24 types of vowels, which are represented by five binary digits. Similarly, the same encoding is used for the front and back nasals. Both consonants and vowels encoding are in Gray Code form, which minimizes the difference between two adjacent encodings with similar pronunciations. The encoding of consonants and vowels is shown in Table 1 and Table 2.

The third part is the additional encoding bit, which is used in the same way as the encoding of the vowel and is represented as five zeros if there is no additional encoding.

The fourth part is the tone, which consists of four tones and can be represented by two binary numbers, 00, 01, 10, and 11 in that order.

Thus, there are a total of 5+5+5+2=17 binary digits in the encoding representation of the phonetic part.

Glyph part: The glyph features of Chinese characters include structure, four-corner coding, and strokes, so these three parts need to be coded.

The first part is the structure of the glyphs. Since Chinese characters have 14 different structures, so they are represented by four binary digits, and the same Gray Code form is used to make the structures with similar glyphs similar in encoding representation. The encoding of glyphs structure is shown in Table 3.

Table 1: Encoding representation of consonants

b=00000	p=00001	m=00011	f=00010
d=00111	t=00101	n=00100	l=01100
g=01111	k=01110	h=01010	
J=01001	q=01000	x=11000	
zh=11011	ch=11010	sh=11110	
z=11011	c=11010	s=11110	r=11111
y=11100	w=10101		

Table 2: Encoding representation of vowels

a=00000	ai=00001	ao=00011	an=00010
i=00111	ie=00101	iu=00100	in=01100
o=01111	ou=01110	ong=01010	ang=11101
e=01001	ei=01000	er=11000	en=11001
u=11011	ui=11110	un=11111	ing=01100
ü=11100	üe=10100	ün=10101	eng=11001

Table 3: Classification accuracy of models

structure	**encoding**	**structure**	**encoding**
single character	0000	Upper right surround	0100
Left-right	0001	Upper three surrounds	1100
Left-center-right	0011	Lower three surrounds	1101
Upper-lower	0010	Left three surrounds	1111
Upper-middle-down	0110	Full surround	1110
Upper Left surround	0111	interpenetration structure	1010
Lower left surround	0101	Structure of the character Pin	1011

The second part is the four-corner coding of Chinese characters, which is used to describe the morphological characteristics of Chinese characters. Each Chinese character can be represented by four numbers from 0 to 9, and the corresponding four-corner coding can be obtained by finding the Four-angle Number Indexing System for Chinese Characters.

The third part is the strokes of the Chinese character, and the number of strokes is expressed as a 16-bit binary number. That is, the number of strokes z is encoded as shown in Equation (6):

$$\begin{cases} 0xFF & \text{if } z > 16 \\ 2^z - 1 & \text{if } z \leqslant 16 \end{cases} \tag{6}$$

Therefore, the encoding representation of the glyph part has a total of 4+ 4× 4+ 16= 36 binary bits.

2) Similarity calculation method
Since the inconsistent number of binary bits in the encoding representation of the phonetic and glyphic parts will have different effects on the final similarity, the contribution ratios b_1 and b_2 of the phonetic

and glyphic parts in the final similarity calculation need to be calculated, satisfying b_1+b_2=1, as shown in Equation (7) and Equation (8), where q_p denotes the Hamming distance of a phonetic encoding, q_x denotes the Hamming distance of a glyph encoding, and l_1 and l_2 denote the binary length of a phonetic and glyph encoding respectively.

$$b_1 = \frac{e^{\frac{q_p}{l_1}}}{e^{\frac{q_p}{l_1}} + e^{\frac{q_x}{l_2}}} \tag{7}$$

$$b_2 = \frac{e^{\frac{q_x}{l_2}}}{e^{\frac{q_p}{l_1}} + e^{\frac{q_x}{l_2}}} \tag{8}$$

The similarity S between Chinese characters is calculated as shown in Equation (9):

$$S = 1 - \frac{q_p}{l_1} \times b_1 - \frac{q_x}{l_2} \times b_2 \tag{9}$$

The similarity is sorted in descending order and the top K characters are added to the adversarial search space.

2.2.2 Word Substitution Strategy

For each word w_i in the original sample x, the word $w_i' \in L_i$ in the adversarial search space L_i corresponding to that word is used for replacement, and the replacement rule is shown in the following:

$$w_i^* = Q\left(w_i, L_i\right) = \underset{w_i' \in L_i}{\operatorname{argmax}} \left\{P\left(y_{\text{true}} \mid \boldsymbol{x}\right) - P\left(y_{\text{true}} \mid \boldsymbol{x}_i'\right)\right\}$$

where $x = w_1 w_2 \ldots w_i \ldots w_n$, $x_i^{'} = w_1 w_2 \ldots w_i^{'} \ldots w_n$, x_i' denotes the text obtained after replacing each word w_i with w_i' in the adversarial search space L_i of w_i, and then replacing w_i with w_i^* to obtain the new text x_i^*.

The change in classification probability between x and x_i^* indicates the best attack that can be achieved after replacing w_i, as shown in Equation (10):

$$\begin{aligned} \boldsymbol{x}_i^* &= w_1 w_2 \ldots w_i^* \ldots w_n \\ \Delta P_i^* &= P\left(y_{\text{true}} \mid x\right) - P\left(y_{\text{true}} \mid x_i^*\right) \end{aligned} \tag{10}$$

2.2.3 Replacement Order Strategy

In the text classification task, each word of the original input text has a different degree of influence on the final classification result, therefore, word saliency is incorporated into the algorithm to determine the replacement order, and the final saliency score $T(x, w_i)$ is shown in Equation (11):

$$T\left(x, w_i\right) = \left\{ \begin{array}{c} P\left(y_{\text{true}} \mid x\right) - P\left(y_{\text{true}} \mid x \backslash w_i\right) \\ \text{if } P(x) = P\left(x \backslash w_i\right) = y_{\text{true}} \\ \left(P\left(y_{\text{true}} \mid x\right) - P\left(y_{\text{true}} \mid x \backslash w_i\right)\right) \\ + \left(P\left(y' \mid x \backslash w_i\right) - P\left(y' \mid x\right)\right) \\ \text{if } P(x) = y_{\text{true}}\, , P\left(x \backslash w_i\right) = y', y' \neq y_{\text{true}} \end{array} \right. \tag{11}$$

where $T(x, w_i)$ as the word saliency score of word $w_i {\in} x$ for the classification result $P(x){=}y_{true}$, $xnw_i{=}w_1 \ldots w_{i-1} w_{i+1} \ldots w_n$ denotes the sentence with word w_i removed, $P(y|x)$ denotes the confidence level of text x predicted as label y. To prioritize the replacement words, the degree of change in the classification probability after replacement and the word saliency of each word was considered. Therefore, the scoring function is determined by ΔP_i^* and $T(x)$, as shown in Equation (12):

$$H\left(x, x_i^*, w_i\right) = \varphi(T(x))_i \cdot \Delta p_i^* \tag{12}$$

where

$$\varphi(z)_i = \frac{e^{z_i}}{\sum_{k=1}^{K} e^{z_k}} \tag{13}$$

$\varphi(z)_i$ is a softmax function that represents the i^{th} component of a vector $\varphi(z)$. And $\varphi(T(x))$ indicates the softmax operation of the word significance vector $T(x)$, z_i represents the i^{th} component of vector z, and K =$|T(x)|$.

The replacement order is determined from the above equation. Arrange all words w in x in descending order according to $H(x, x_i^*, w_i)$. Then consider each word w_i under that order and replace it with w_i^* and repeat the process until the classification label is changed and the attack is successful.

The Word substitution and Replacement order strategy algorithm is shown in Algorithm 1.

Algorithm 1 The Word substitution and Replacement order strategy algorithm

```
Input:
    Text $x^{(0)}$ before iteration;
    Length of text $x^{(0)}$: $n$ =$|x^{(0)}|$;
    classifier $F$; Modify threshold $\sigma$;
Output:
    Adversarial example $x^{(i)}$
Begin
$x^{(0)} = w_1 w_2 ... w_i ... w_n$
for all $i = 1$ to $n$ do
    Compute word saliency $T(x^{(0)}, w_i)$
    Get a set $L_i$ for $w_i$
    Replace $w_i$ with $w_i^{'} \in L_i$
    Get $x_i^{'} = w_1 w_2 \ldots w_i^{'} \ldots w_n$
    $w_i^* = Q(w_i, L_i)$
    Get $x_i^* = w_1 w_2 \ldots w_i^* \ldots w_n$
    Compute $\Delta P_i^*$ according to Eq.11
end for
Reorder $w_i$ such that
$H(x,x_1^*,w_i)$¿ . . . ¿ $H(x,x_n^*,w_n)$
for all $i = 1$ to $n$ do
    Replace $w_i$ in $x^{(i-1)}$ with $w_i^*$ to craft $x^{(i)}$
    if $F(x^{(i)}){\neq}F(x^{(0)})$ and $Cost(x^{(i)}, x^{(0)}){\leqslant}\sigma$ then
        return $x^{(i)}$
    end if
end for
```

2.2.4 Optimize the Adversarial Example

A good adversarial example should not only achieve a high attack success rate but also maintain a certain degree of semantic similarity with the original sample, i.e., it can induce the model to make false discriminations and also make it undetectable to humans.

Therefore, under the premise that the generated examples are adversarial, in order to improve the similarity between the original samples and the adversarial examples and enhance the semantic consistency, w_i^* in the adversarial examples $x^{(i)}$ is replaced in turn by finding words from the adversarial search space L_i of their original words w_i to satisfy Equation (14):

$$\max_{x^{(i)}} S\left(x, x^{(i)}\right) \quad \text{s.t.} \quad C\left(\mathrm{P}\left(x^{(i)}\right)\right)=1 \tag{14}$$

where C is an adversarial criterion that equals 1 if $x^{(i)}$ is an adversarial example and 0 otherwise. $S(x, x^{(i)})$ denotes the semantic similarity between the original sample x and the adversarial example $x^{(i)}$, measured by the cosine value. If the replacement does not satisfy the adversarial criterion then it is output as the final adversarial example.

3 Experimental Evaluation

3.1 Dataset

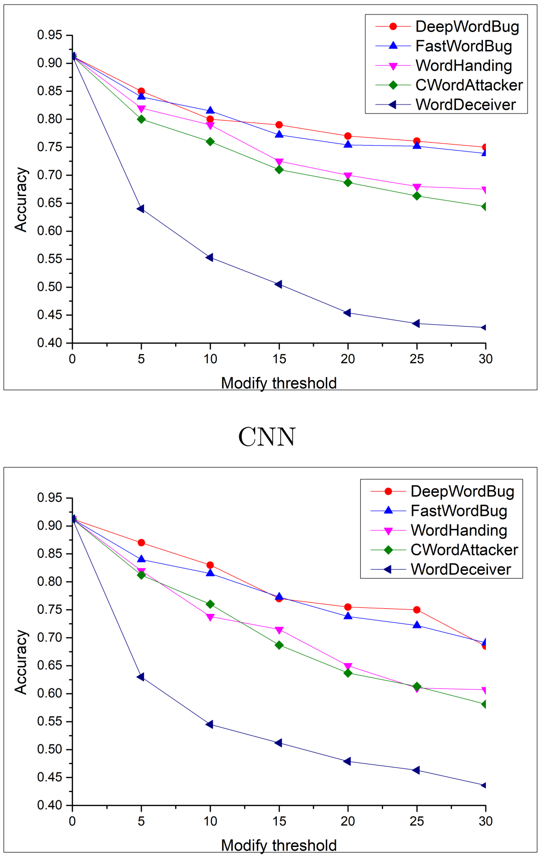
Four Chinese datasets were selected for evaluation, as shown in Table 4.

3.2 Target Model

The CNN consists of a 300-dimensional embedding layer, three convolutional layers, and a fully connected layer. The convolutional layer consists of 256 convolutional kernels of sizes 2,3,4 with a step size of 1. The BiLSTM consists of a forward LSTM and a backward LSTM, a 300-dimensional embedding layer, and a fully connected layer.

3.3 Comparison of Experimental Methods

To verify the relationship between the accuracy of adversarial example detection and the modification threshold m, 1000 examples with lengths greater than 120 words were selected from each of the two sentiment classification datasets, and the corresponding adversarial examples were generated by adjusting the different modification magnitudes. Four attack algorithms were compared with DeepWordBug, FastWordBug, WordHanding, and CWordAttacker on the sentiment analysis dataset. Respectively, setting the parameter K=35. The maximum modification threshold allowed for the Ctrip hotel, Jingdong shopping, and spam datasets was 30, and the maximum modification threshold for the news classification was 11.



CNN

BiLSTM

Figure 1: The variation curve of detection accuracy with modification threshold for the adversarial example of the Ctrip review dataset

The variation curves of detection accuracy with modification threshold m for CNN and BiLSTM on the Ctrip hotel dataset and Jingdong shopping review dataset are shown in Figure 1 and Figure 2.

As the modification threshold m increases, the detection accuracy of the model gradually decreases, implying that WordDeceiver can generate adversarial examples by making modifications to individual keywords in the input sequence. Compared with the baseline, the attack effect tends to a steady state when the modification threshold reaches about 18 words, indicating that the WordDeceiver algorithm greatly reduces the perturbation rate and improves the readability of the text.

To verify the effectiveness of the WordDeceiver algorithm, four attack algorithms were compared with DeepWordBug, FastWordBug, WordHanding, and CWordAttacker on the sentiment analysis dataset. The experimental results are shown in Tables 5 and 6.

The effect of the attack on CNN and BiLSTM models was analyzed on four datasets. As can be seen from Table 5, on the sentiment analysis dataset compared to the baseline approach, the WordDeceiver algorithm proposed in this paper can reduce the accuracy by up to 48.46%,

Table 4: Classification accuracy of models

Dataset	Task	Classes	Train	Test	Average words
Ctrip	Sentiment analysis	2	12000	3000	135
JD	Sentiment analysis	2	35000	5000	32
Spam	Spam classification	2	90000	10000	56
THUCNews	News classification	10	90000	10000	23

Table 5: Validation of the algorithm WordDeceiver on sentiment analysis tasks

(a) CNN

Dataset	Method	Ori_acc(%)	Accuracy(%)	Reduction
Ctrip	DeepWordBug	91.27	75.46	15.81
	FastWordBug		73.91	17.36
	WordHanding		67.24	24.03
	CWordAttacker		66.47	24.80
	WordDeceiver		42.81	48.46
JD	DeepWordBug	90.85	72.62	18.23
	FastWordBug		71.25	19.60
	WordHanding		68.16	22.69
	CWordAttacker		68.02	22.83
	WordDeceiver		49.58	41.27

(b) BiLSTM

Dataset	Method	Ori_acc(%)	Accuracy(%)	Reduction
Ctrip	DeepWordBug	91.26	68.52	22.74
	FastWordBug		69.15	22.11
	WordHanding		60.77	30.49
	CWordAttacker		58.19	33.07
	WordDeceiver		43.61	47.65
JD	DeepWordBug	93.04	70.28	22.76
	FastWordBug		69.54	23.50
	WordHanding		63.81	29.23
	CWordAttacker		61.96	31.08
	WordDeceiver		51.19	41.85

which is significantly better than the baseline. As can be seen from Table 6, the WordDeceiver algorithm on the spam and news headline datasets can reduce the accuracy of the model by about 40% on average, demonstrating its effectiveness and versatility in multi-scenario tasks.

3.4 Adversarial Examples Quality Measurement

WMD (Word Mover's Distance) is used to measure the distance between two text documents, which is used to determine the similarity between two texts, i.e., the larger the WMD distance the smaller the text similarity, and the smaller the WMD distance the larger the similarity. From the generated adversarial examples, 2000 examples were randomly selected for the experiment, and the experimental results are shown in Figure 3.

Compared with the baseline, the adversarial examples with WMD distance in the range of 0-0.2 account for 33.8% of the total number of experimental samples, which have higher similarity with the original samples; the total percentage of adversarial examples in the interval of 0-0.6 is 75.1%, which is higher than the baseline method, indicating that the adversarial examples generated by the WordDeceiver algorithm have less semantic deviation and higher similarity with the original samples. Table 7 shows an example of the original samples and the adversarial examples generated by the WordDeceiver algorithm. It can be found that the generated adversarial examples can still be understood by the semantic context, which preserves the semantics well and is highly readable.

3.5 Transferability Assessment

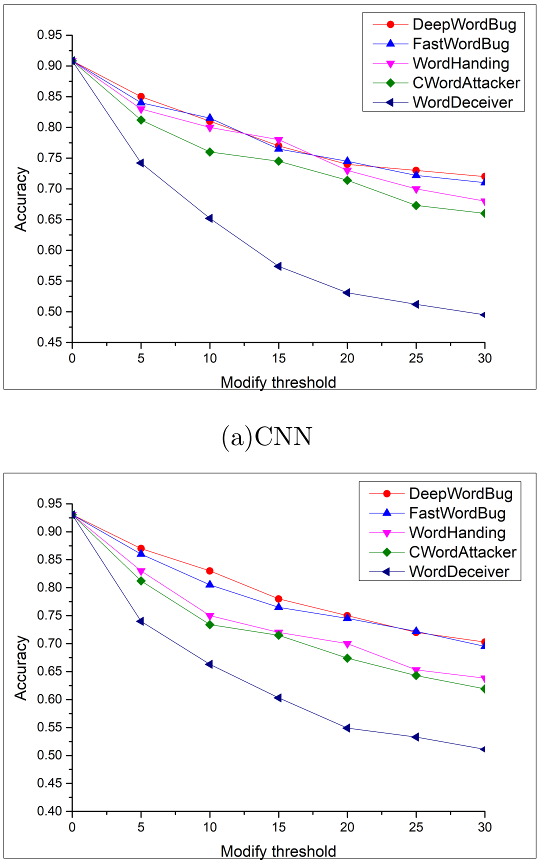
To verify that the adversarial examples generated by the WordDeceiver algorithm are transferable, experiments us-

Table 6: Evaluating WordDeceiver performance on spam and THUCNews classification tasks

Dataset	Model	Ori_acc(%)	CWordAttacker		WordDeceiver	
			Accuracy(%)	Reduction	Accuracy(%)	Reduction
Spam	CNN	99.91	87.67	12.24	53.44	46.47
	BiLSTM	99.84	87.25	12.59	53.23	46.61
THUCNews	CNN	91.52	68.90	22.62	57.13	34.39
	BiLSTM	91.23	62.49	28.74	56.58	34.65

Table 7: Examples of original samples and generated adversarial examples

Original sample	Label	Adversarial example	Label
服务非常好，环境也很舒适，只是酒店的餐饮方面一般般。	Positive	服务诽尝奶，环境也很纾括，只是酒店的餐饮方面一般般。	Negative
洗一段时间后，头皮屑严重，最糟糕的是，头痒的难受。	Negative	洗一段时间后，头皮屑彦锺，最槽恙的是，头痒的滩爱。	Positive



(a)CNN

(b)BiLSTM

Figure 2: The variation curve of detection accuracy with modification threshold for the adversarial example of the JD review dataset

ing BiLSTM and CNN models to generate adversarial examples to attack other models (including the BERT [2] model) and Cloud Platforms are shown in Table 8 and Table 9, respectively. The experimental results show that the adversarial examples generated by the WordDeceiver algorithm can be effectively transferred to other models and Cloud Platforms with an accuracy reduction of about 30%.

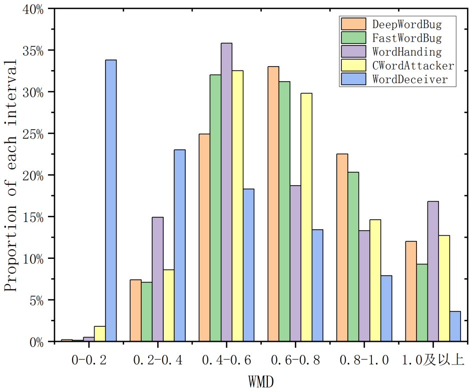


Figure 3: Proportion of the number of examples in different WMD distance intervals to the total examples

3.6 Human Evaluation

We performed a human evaluation of the adversarial examples generated by the WordDeceiver algorithm. Three main aspects were evaluated: accuracy, the naturalness of the adversarial examples from a perceptual perspective (N_{score}), and similarity (S_{score}). The researchers randomly selected 200 original samples from both sentiment analysis datasets and their corresponding adversarial examples to be classified by volunteers and rated them on a scale of 1 to 5, with higher ratings meaning better quality of the generated adversarial examples. A total of 10 volunteers participated in the experiment, and the evaluation results are shown in Table 10. As can be seen from Table 10, the human classification effect is still good. The difference between the classification accuracy of the adversarial example and the original sample is less than 3%, and the naturalness score and similarity score are above 4. This indicates that the adversarial example generated by the WordDeceiver algorithm retains the semantics to a large extent and can be better understood by humans.

Table 8: Results of adversarial examples generated using BiLSTM model to attack other models and Cloud Platforms

Dataset	Model/Cloud Platform	Ori_acc(%)	WordDeceiver	
			Accuracy(%)	Reduction
Ctrip	CNN	91.27	57.85	34.42
	BERT	90.13	58.12	32.01
	Tencent Cloud	85.64	59.97	25.67
	Baidu AI	87.08	61.56	25.52
JD	CNN	90.85	60.98	29.87
	BERT	91.24	61.85	29.39
	Tencent Cloud	88.27	63.89	24.38
	Baidu AI	87.35	64.06	23.29

Table 9: Results of adversarial examples generated using CNN model to attack other models and Cloud Platforms

Dataset	Model/Cloud Platform	Ori_acc(%)	WordDeceiver	
			Accuracy(%)	Reduction
Ctrip	CNN	91.26	59.98	34.28
	BERT	90.13	60.60	31.53
	Tencent Cloud	85.64	59.32	26.32
	Baidu AI	87.08	60.92	26.16
JD	CNN	93.04	63.57	30.47
	BERT	91.24	64.32	27.92
	Tencent Cloud	88.27	61.51	26.76
	Baidu AI	87.35	61.94	25.41

3.7 Adversarial Training

Adversarial training refers to the method of constructing an adversarial example during the training of the model and mixing the adversarial example with the original sample to train the model, in other words, adversarial attacks are performed on the model during the training process to improve the robustness of the model against adversarial attacks. Therefore, in order to improve the robustness of the model, 5000 original samples are randomly selected from the Ctrip hotel review data, the corresponding adversarial examples are generated on the BiLSTM model, and a number of adversarial examples are randomly selected and added to the original sample training set for training, and the original test set and the adversarial example test set are evaluated.

As can be seen from Figure 4, the more adversarial examples are added to the training set, the higher the accuracy of classification is and reaches more than 80%, indicating that adversarial training can effectively improve the robustness of the model.

4 Conclusion

In this paper, we propose WordDeceiver, an adversarial attack method for classifying Chinese text under black-box conditions, which can induce the model to make wrong decisions. The method first constructs adversarial search space for each Chinese character by combining the character glyph and phonetic features, determines the replacement order by word salience and classification probability, generates adversarial examples using word Substitution strategy, and designs a new method to improve the semantic similarity between the adversarial examples and the original samples while ensuring the success of the attack. The adversarial examples generated by the WordDeceiver algorithm can reduce the accuracy of CNN and BiLSTM models by up to 48.46% and 46.61%, and transfer to other models with an accuracy reduction of about 30%, and the attacks are all better than other attack methods. In addition, the word importance calculation function and modification strategy can be further optimized and improved. In future work, we will analyze and improve these problems and conduct more in-depth exploration and research on how to enhance the robustness of text classification models.

Acknowledgments

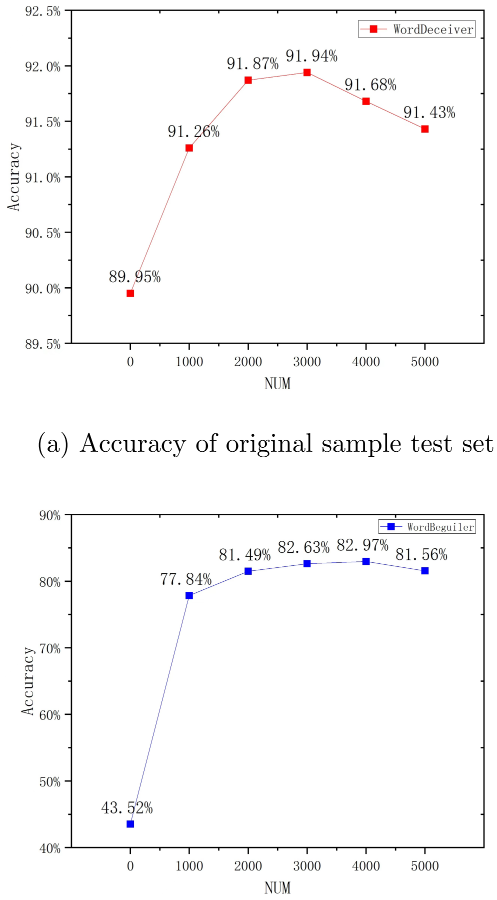
This work was partially supported by the Natural Science Foundation of China (No.61562065) and the Natural Science Foundation Project of Inner Mongolia (No.2019MS06001 and No.2019MS06036). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *IEEE Symposium on Security and Privacy (SP'17)*, 2017.

Table 10: Comparison with human evaluation

Dataset	Model	Examples	Acc_model(%)	Acc_huma(%)	N_{score}	S_{score}
Ctrip	*CNN*	Original	98.00	97.00	4.80	5.00
		Adversarial	17.00	96.00	4.05	4.65
	BiLSTM	Original	96.00	97.00	4.80	5.00
		Adversarial	20.00	96.00	4.16	4.60
JD	*CNN*	Original	97.00	98.00	4.75	5.00
		Adversarial	21.00	95.00	4.25	4.60
	BiLSTM	Original	96.00	98.00	4.75	5.00
		Adversarial	24.00	96.00	4.30	4.65



(a) Accuracy of original sample test set

(b) Accuracy of adversarial example test set

Figure 4: Effect of adversarial training on the classification accuracy of original and adversarial examples

[2] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[3] Y. Feng, C. Hu, H. Kamigaito, H. Takamura, and M. Okumura, "Improving character-aware neural language model by warming up character encoder under skip-gram architecture," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP'21)*, pp. 421–427, 2021.

[4] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, "Black-box generation of adversarial text sequences to evade deep learning classifiers," in *IEEE Security and Privacy Workshops (SPW'18)*, pp. 50–56, 2018.

[5] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *Computer Science*, 2014.

[6] Dou Goodman, Lv Zhonghou, et al., "Fastwordbug: A fast method to generate adversarial text against nlp applications," *arXiv preprint arXiv:2002.00760*, 2020.

[7] Kathrin Grosse, Nicolas Papernot, Praveen Manoharan, Michael Backes, and Patrick McDaniel, "Adversarial examples for malware detection," in *European symposium on research in computer security*, pp. 62–79. Springer, 2017.

[8] Bojan Kolosnjaji, Apostolis Zarras, George Webster, and Claudia Eckert, "Deep learning for classification of malware system call sequences," in *Australasian joint conference on artificial intelligence*, pp. 137–149. Springer, 2016.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[10] Tzu-Hsuan Lin, Yu-Hua Huang, and Alan Putranto, "Intelligent question and answer system for building information modeling and artificial intelligence of things based on the bidirectional encoder representations from transformers model," *Automation in Construction*, vol. 142, p. 104483, 2022.

[11] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 142–150, 2011.

[12] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017.

[13] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao, "Deep learning–based text classification: a

comprehensive review," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1–40, 2021.

[14] Nicolas Papernot, Patrick McDaniel, Ananthram Swami, and Richard Harang, "Crafting adversarial input sequences for recurrent neural networks," in *MILCOM 2016-2016 IEEE Military Communications Conference*, pp. 49–54. IEEE, 2016.

[15] A Rakhlin, "Convolutional neural networks for sentence classification," *GitHub*, 2016.

[16] Irene Rivera-Trigueros, "Machine translation systems and quality assessment: a systematic review," *Language Resources and Evaluation*, vol. 56, no. 2, pp. 593–619, 2022.

[17] Alireza Sadeghi, Hamid Bagheri, Joshua Garcia, and Sam Malek, "A taxonomy and qualitative comparison of program analysis techniques for security assessment of android software," *IEEE Transactions on Software Engineering*, vol. 43, no. 6, pp. 492–530, 2016.

[18] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *Computer Science*, 2013.

[19] Xin Tong, Luona Wang, Runzheng Wang, and Jingya Wang, "A generation method of word-level adversarial samples for chinese text classification," *Netinfo Secur*, vol. 20, no. 09, pp. 12–16, 2020.

[20] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[21] WQ Wang, R Wang, LN Wang, and BX Tang, "Adversarial examples generation approach for tendency classification on chinese texts," *Ruan Jian Xue Bao/J. Softw.*, vol. 30, no. 8, pp. 2415–2427, 2019.

[22] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria, "Recent trends in deep learning based natural language processing," *ieee Computational intelligenCe magazine*, vol. 13, no. 3, pp. 55–75, 2018.

[23] Dong Yu and Li Deng, *Automatic speech recognition*, vol. 1. Springer, 2016.

[24] Lei Zhang, Shuai Wang, and Bing Liu, "Deep learning for sentiment analysis: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1253, 2018.

[25] Hong Zhao, You kang Chang, and Wei jie Wang, "Research on robustness of deep neural networks based data preprocessing techniques," *International Journal of Network Security*, vol. 24, no. 2, pp. 243–252, 2022.

Biography

Yan Gong was born in huhhot, china, in November 1997.She is a graduate student in the School of Information Engineering, Inner Mongolia University of Science & Technology. Her research interests include adversarial attacks based on text classification.

Xiao-Lin Zhang was born in December 1966 in Baotou, China. She received a B.S. in Computer Science and Technology from Northeastern University in 1988, an M.S. in Automation from Beijing University of Science and Technology in 1995, and a Ph.D. in Computer Science and Technology from Northeastern University in 2006. Since 1988, she has been working at Inner Mongolia University of Science and Technology, where she is currently the Deputy Head of the Computer Science Department, Professor in the College of Information Technology, and Head of the Computer Science Department. She has trained more than 70 master students and is currently training 12 master students. She has published more than 80 scientific papers, including more than 30 papers in EI and 7 papers in SCI. She is responsible for many projects, including the National Natural Science Foundation of China, the National Social Science Fund Project, the Natural Science Foundation of Inner Mongolia Project, the Chunhui Project of the Ministry of Education, and the Inner Mongolia Education Department Fund Project. Her current research interests include image processing, natural language processing, image and text attacks, machine learning security, big data processing technology, and social network privacy technology. Dr. Zhang is a member of the Chinese Computer Society, the Professional Committee of Information Systems, the Chinese Computer Society, and the Director of the Computer Society of Inner Mongolia Autonomous Region.

Yong-Ping Wang was born in Baotou, China, in 1984. She graduated from Wuhan Technical University with a Bachelor's degree in Computer Science and Technology in 2007 and a Master's degree in 2010. Since 2010, she has been with the Inner Mongolia University of Science and Technology, where she is currently a lecturer with the School of Information Engineering. She has participated in a number of research projects of the Inner Mongolia Natural Science Foundation. Her main research interests include image processing, machine learning security, and adversarial attacks on images.

Li-Xin Liu was born in Baotou, China, in 1984. She received a bachelor's degree in information security from Central South University, in 2007, and a master's degree in computer science and technology from Central South University, in 2010. She received her Ph.D. degree at the Renmin University of China in 2021. Since 2010, she has been with the Inner Mongolia University of Science and Technology, where she is currently a Lecturer with the Department of Computer Science, School of Information Engineering. She has presided over one provincial and ministerial-level scientific research project, one school-level project, four scientific research projects, and four academic articles, including one EI journal. Her main research interests include privacy protection and blockchain, machine learning security, image processing, and adversarial attacks on images.

En-Hui Xu was born in Jiangsu, china, in 1997. He graduated from the Inner Mongolia University of Science and Technology in 2022 with a Master's degree and worked at the China Nanhu Academy of Electronics and Information Technology in July of the same year. He published one EI paper and one SCI paper during his master's degree. His research interests include machine learning security and natural language processing.

Secure Multiparty Multisets Computation

Jiahao Pan and Jiawei Dou
(Corresponding author: Jiawei Dou)

School of Mathematics and Statistics, Shaanxi Normal University
Xi'an 710119, China
Email: jiawei@snnu.edu.cn

(Received Oct. 5, 2022; Revised and Accepted Mar. 3, 2023; First Online Apr. 27, 2023)

Abstract

Secure multiparty computation (SMC) is an important technology to protect data privacy in cooperative computations and has developed into a hot research field in cryptography. Private set computation is an important aspect of SMC, and secure multisets computation is essential in practice. Privately computing the sum of elements of the intersection of multisets is a new secure multiparty computation problem that has not been studied, which also has many applications in practice. This paper addresses this intersection sum problem. First, we design a new encoding scheme, one participant encodes their private multiset as a matrix, and the other participants perform some operations on the matrix according to their multiset in turn and finally obtain the sum. We use a threshold elliptic curve cryptosystem to design a protocol to realize this process and use the well-accepted simulation paradigm to strictly prove that the protocol is secure in the semi-honest model. Theoretical analysis and experimental results show that our protocol is highly efficient.

Keywords: Cryptography; Elliptic Curve Cryptosystem; Secure Multiparty Computation

1 Introduction

Yao [21] first proposed and studied the millionaire problem, which pioneered the secure two-party computation. After that, Goldwasser and Goldreich, *et al.* [3, 4] proposed and studied the general secure multiparty computation(SMC) problem. Now, SMC has become a very hot research topic in cryptography, involving range query [13,17], location query [5,22], set problems [15,18], vector problems [11,19].

Multisets are widely used in life. For example, multiset is used to record the information about age and occupation collected by banks, blood type, blood pressure, and heart rate of patients collected by a hospital. Such information is private information. If multi parties use these data to perform cooperative computation, a secure multiparty computation protocol is necessary.

There has been some research work on secure multiset computation problems. [12] used the polynomial oblivious computation to design the protocol of multiset intersection, union, and their cardinalities. [1] used oblivious sorting and date comparison to design protocols for standard sets and multisets operations. [2] used matrix and homomorphic encryption schemes to address secure two-party multiset intersection computation.

The elements sum of intersection or subset also are important set operation problems, and there are few research results on such problems [14,16]. [16] addressed the cardinality and the sum of the elements of the intersection of two private standard sets. In this paper, we study the secure multiparty computation of the sum of elements of the intersection of some private multisets. The main contribution of this paper is as follows:

1) We design an encoding scheme that encodes a multiset into a matrix and computes the sum of elements in the matrix to compute the sum of the elements of the intersection of some multisets.

2) Because the threshold elliptic curve cryptosystem (TECC) has additive homomorphism, and can resist the collusion attack, we apply TECC to design a protocol to realize the private computation of sum of elements of the intersection.

3) We analyze and test the efficiency of the protocol. The theoretical analysis and experimental results show that the proposed protocol is efficient and feasible.

2 Preliminaries

2.1 Semi-honest Models and Security

In this model, participants will follow the protocol as required, but keep the information received during the execution of the protocol, and try to derive additional information about other participants' private data after the execution.

Assume that there are n participants $P_1, \cdots, P_n$, having private input data $x_1, \cdots, x_n$, respectively. Denote

$\bar{x} = (x_1, \cdots, x_n)$. The parties want to compute the function $f(\bar{x}) = (f_1(\bar{x}), \cdots, f_n(\bar{x}))$ through protocol π and $P_i, i \in [1, n] = \{1, \cdots, n\}$ receives output $f_i(\bar{x})$. During the execution of the protocol, P_i receives message series $view_i^\pi(\bar{x}) = (x_i, r_i, M_i^1, \cdots, M_i^{t_i}, f_i(\bar{x}))$, where M_i^j represents the j-th message P_i received. r_i is the random number P_i chosen. Suppose $I = \{P_{i_1}, \cdots, P_{i_s}\} \subset \{P_1, \cdots, P_n\}$. $view_I^\pi(\bar{x})$ is defined as

$$view_I^\pi(\bar{x}) = (I, view_{i_1}^\pi(\bar{x}), \cdots, view_{i_s}^\pi(\bar{x})).$$

Definition 1. *If for any subset $I \subset \{P_1, \cdots, P_n\}$, there is a probabilistic polynomial-time algorithm S such that*

$$\{S(I, (x_{i_1}, \cdots, x_{i_s}, f_I(\bar{x}))\} \overset{c}{\equiv} \{view_I^\pi(\bar{x})\}, \qquad (1)$$

where $f_I(\bar{x}) = (f_{i_1}(\bar{x}), \cdots, f_{i_s}(\bar{x}))$, and $\overset{c}{\equiv}$ denotes computationally indistinguishable, we say that protocol π securely computes function $f(\bar{x})$ in the semi-honest model.

2.2 Threshold Elliptic Curve Cryptosystem

The elliptic curve cryptosystem is an important public key cryptosystem with additive homomorphism and has important applications in SMC [6–10, 20]. In this paper, we apply the threshold elliptic curve cryptosystem (TECC) to design our protocol. Three algorithms for a TECC are described below:

Key generation. All participants $P_1, \cdots, P_n$ cooperatively choose an elliptic curve $E_p(a, b)$ and a base point G on the curve, where p is a large prime number, and request the order of G is a large enough prime number (the order of G is the minimum positive integer L such that $LG = O$). Each P_i chooses a random number k_i as his/her private key, and computers $K_i = k_i G$. The public key is $K = \sum_{i=1}^n K_i$.

Encryption. A plaintext message m is first encoded to a point M on the $E_p(a, b)$, choose a random number $r(1 \le r \le L - 1)$, and computes the ciphertext $E(m) = (C_1, C_2) = (M + rK, rG)$.

Cooperation decryption. To decrypt a ciphertext $C = (C_1, C_2)$, the participant P_i calculates $k_i C_2$ and publishes it (this process is called partial decryption). Participants further calculate $M = C_1 - \sum_{i=1}^n k_i C_2$, and then decode M to get the final decryption result $m = D(C)$ (this process is called fully decryption).

Remark 1. *In the following, we encode m as mG where G is a base point of the elliptic curve, which keeps the additive homomorphism of the cryptographic scheme to the original plaintext message. In this way, coding m_1, m_2 to m_1G, m_2G, $E(m_1) + E(m_2) = E(m_1 + m_2)$, that is, the ciphertext of $(m_1 + m_2)$ can be directly calculated from the ciphertexts of m_1 and m_2 without decryption. When decrypting (C_1, C_2), only $(m_1 + m_2)G$ can be obtained, due to the difficulty of the discrete logarithm problem on elliptic curves, $m_1 + m_2$ cannot be obtained directly from $(m_1 + m_2)G$, and further decoding is required to obtain $m_1 + m_2$. However, when the range of m_1 and m_2 is not too large, a coding table can be made, and the value of $m_1 + m_2$ can be determined by looking up the table.*

Because in a threshold cryptography system, a ciphertext can be decrypted only when some participants work together, the threshold cryptography system is an important mean to resist collusion attacks. In the following, we use TECC to design our protocol, and all the operations in the protocol are performed on the elliptic curve group $E_p(a, b)$.

2.3 Re-randomization and Semantic Security of TECC

Because the elliptic curve cryptosystem has additive homomorphism, by adding an $E(0)$ to $E(m)$, a new ciphertext of m can be obtained, and this process is called rerandomization.

TECC is semantically secure, which means the same plaintext can be encrypted into different ciphertext forms, and all ciphertexts are computationally indistinguishable.

3 Securely Computing the Sum of Elements of the Intersection of Multisets

3.1 Problem Describe

Suppose P_i ($i \in [1, n]$) owns private multiset X_i, the elements of these multisets belong to set $Q = \{q_1, \cdots, q_l\}$. For any $i \in [1, n], k \in [1, l]$, let $s_k^{(i)}$ represent the times that q_k appears in X_i, and suppose that $s_k^{(i)}$ does not exceed a positive integer m. Then X_i can be written as $X_i = \{(q_k, s_k^i)\}_{k=1}^l$, and $X = \bigcap_{i=1}^n X_i = \{(q_k, \alpha_k)\}_{k=1}^l$ represents the intersection of $X_1, \cdots, X_n$, where α_k represents the times that q_k appears in X, with $\alpha_k = \min\{s_k^{(1)}, \cdots, s_k^{(n)}\}$.

$P_1, \cdots, P_n$ want to compute the sum of elements of the intersection X (i.e., $\sum_{k=1}^l \alpha_k q_k$) without revealing any additional information about the private multiset X_i.

For convenience, we set $\bar{X} = (X_1, \cdots, X_n)$, and define the function $y = F(\bar{X}) = \sum_{k=1}^l \alpha_k q_k$. In the following, we will design the secure computation protocol for $y = F(\bar{X})$.

3.2 Encoding Scheme and Protocol Design

Encoding scheme 1. In the protocol design, the participant P_1 first encodes his/her private multiset X_1 into an $l \times m$ matrix $A^{(1)}$: for any $k \in [1, l]$, the k-th row of the matrix $A^{(1)}$ is expressed as $A_k^{(1)} = (a_{k1}^{(1)}, \cdots, a_{km}^{(1)})$ in which there are $s_k^{(1)}$ q_k's followed by $m - s_k^{(1)}$ zeros, that is,

$$A_k^{(1)} = (\overbrace{q_k, \cdots, q_k}^{s_k^{(1)}}, \overbrace{0, \cdots, 0}^{m-s_k^{(1)}}). \tag{2}$$

Protocol 1 Privately computing the sum of the elements of the intersection of n multisets.

Input: $X_i = \{(q_k, s_k^{(i)})\}_{k=1}^{l}, i \in [1, n]$.

Output: $y = F(\bar{X}) = F(X_1, \cdots, X_n)$.

Setup: $P_1, \cdots, P_n$ jointly select an elliptic curve $E_p(a, b)$ and a base point G on the curve. P_i selects k_i as private key and jointly generate public key $pk = \sum_{i=1}^{n} k_i G$.

1) P_1 does the following:

   a. Encodes X_1 into $A^{(1)} = (a_{kj}^{(1)})_{l \times m}$ according to Encoding scheme 1.

   b. Uses the public key pk to encrypt each element of $A^{(1)}$ and obtains $C^{(1)} = E(a_{kj}^{(1)})_{l \times m}$, and sends $C^{(1)}$ to P_2.

2) For $i = 2$ to $n - 1$

   For each $k \in [1, l]$, P_i replaces $m - s_k^{(i)}$ elements of the k-th row in $C^{(i-1)}$ with $E(0)$ from back to front and randomizes the other elements of the k-th row. Denotes the new matrix by $C^{(i)}$ and sends $C^{(i)}$ to P_{i+1}.

3) P_n obtains $C^{(n-1)} = (c_{kj}^{(n-1)})_{l \times m}$, and computes

$$C = \sum_{k=1}^{l} (\sum_{j=1}^{s_k^{(n)}} c_{kj}^{(n-1)}) + E(0).$$

4) $P_i, i \in [1, n]$ partially decrypts C using private key share k_i to get Y_i, and publishes Y_i. All parties can calculate the full decryption result y. Outputs y.

4 Security Proof

Theorem 1. *Protocol 1 is secure in the semi-honest model and can resistant any collusion attack.*

Proof. Since in Protocol 1, the status of P_1 and P_n are different from that of $P_2, \cdots, P_{n-1}$ while the status of $P_2, \cdots, P_{n-1}$ are equal. Therefore, it is suffice to prove that X_1, X_2, X_n are secure. For X_i, the most serious attack is that all other parties collude to obtain the information about X_i, therefore, it suffices to prove that there exists a simulator S for $I_i = \{P_1, \cdots, P_n\} \backslash \{P_i\}$ such that Formula (1) holds.

Case 1. X_1 is secure. We prove this by constructing simulator S such that Formula (1) holds. S works as follows:

1) Given input $(I_1, X_2, \cdots, X_n, F(\bar{X}))$, S randomly chooses an X_1' such that $F(X_1', X_2, \cdots, X_n) = F(\bar{X})$.

2) S generates the private key shares $k_i', i \in [1, n]$, and the public key is pk'. S first encodes X_1' as $\hat{A}^{(1)} = (\hat{a}_{kj}^{(1)})_{l \times m}$ following Encoding scheme 1, and encrypts each element of $\hat{A}^{(1)}$ with pk' to obtain $\hat{C}^{(1)}$.

3) For $i = 2, \cdots, n$, S simulates step 2) of Protocol 1, obtains $\hat{C}^{(n-1)} = (\hat{c}_{kj}^{(n-1)})_{l \times m}$, and computes

$$\hat{C} = \sum_{k=1}^{l} (\sum_{j=1}^{s_k^{(n)}} \hat{c}_{kj}^{(n-1)}).$$

4) S first applies the private key shares $k_i', i \in [1, n]$ to partially decrypt $\hat{C}$, gets $\hat{Y}_1, \hat{Y}_2, \cdots, \hat{Y}_n$, and obtains the full decryption result $F(X_1', X_2, \cdots, X_n)$.

In the execution of Protocol 1,

$$view_{I_1}^{\pi}(\bar{X}) = \{X_2, \cdots, X_n, C^{(1)}, Y_1, F(\bar{X})\}.$$

Set

$$\begin{aligned} & S(X_2, \cdots, X_n, F(X_1, X_2, \cdots, X_n)) \\ & = \{X_2, \cdots, X_n, \hat{C}^{(1)}, \hat{Y}_1, F(X_1', X_2, \cdots, X_n)\}. \end{aligned}$$

Because TECC is semantically secure, all the elements in $C^{(1)}$ are computationally indistinguishable from the elements of $\hat{C}^{(1)}$, and the partial decryption result Y_1 is also computationally indistinguishable from $\hat{Y}_1$. Because $F(\bar{X}) = F(X_1', X_2, \cdots, X_n)$, therefore,

$$\begin{aligned} & \{S(I_1, X_2, \cdots, X_n, F(X_1', X_2, \cdots, X_n))\}_{X_1} \\ & \overset{c}{\equiv} \{view_{I_1}^{\pi}(X_1, X_2, \cdots, X_n)\}_{X_1}. \end{aligned}$$

Case 2. X_n is secure. Simulator S for I_n works as follows:

1) Given input $(I_n, X_1, \cdots, X_{n-1}, F(\bar{X}))$, S randomly chooses an X_n' (suppose that the times of q_k appears in X_n' is $\hat{s}_k^{(n)}$), such that $F(\bar{X}) = F(X_1, \cdots, X_{n-1}, X_n')$.

2) S generates the private key shares $k_i', i \in [1, n]$, and the public key is pk'. S uses pk' to encrypt each element of $A^{(1)}$, and obtains the ciphertext matrix $\hat{C}^{(1)}$.

3) For $i = 1, \cdots, n-1$, S simulates step 2) of Protocol 1, and obtains $\hat{C}^{(n-1)} = (\hat{c}_{kj}^{(n-1)})_{l \times m}$.

4) According to the X_n', S computes:

$$\hat{C} = \sum_{k=1}^{l} (\sum_{j=1}^{\hat{s}_k^{(n)}} \hat{c}_{kj}^{(n-1)}).$$

5) S first applies the private key shares $k_i', i \in [1, n]$ to partially decrypt $\hat{C}$ to get $\hat{Y}_i, i \in [1, n]$, and obtains the full decryption result $F(X_1, \cdots, X_{n-1}, X_n')$.

In the execution of Protocol 1,

$$view^{\pi}_{I_n}(\bar{X}) = \{X_1, \cdots, X_{n-1}, C, Y_n, F(\bar{X})\}.$$

Let

$$\begin{aligned} & S(I_n, X_1, \cdots, X_{n-1}, F(\bar{X}_1, \cdots, X_{n-1}, X_n)) \\ & = \{X_1, \cdots, X_{n-1}, \hat{C}, \hat{Y}_n, F(X_1, \cdots, X_{n-1}, X'_n)\}. \end{aligned}$$

Since P_n adds a ciphertext $E(0)$ when calculating C, according to the semantic security of TECC, even if all other participants collude, no information about X_n be leaked, then we have $C \overset{c}{\equiv} \hat{C}$, and the partial decryption result Y_n is also computationally indistinguishable from $\hat{Y}_n$. And because $F(\bar{X}) = F(X_1, \cdots, X_{n-1}, X'_n)$, therefore,

$$\begin{aligned} & \{S(I_n, X_1, \cdots, X_{n-1}, F(X_1, \cdots, X_{n-1}, X_n))\}_{X_n} \\ & \overset{c}{\equiv} \{view^{\pi}_{I_n}(X_1, \cdots, X_{n-1}, X_n)\}_{X_n}. \end{aligned}$$

Case 3. X_2 is secure. Similar to the Case 2, we can also construct a simulator S for I_2, such that Formula (1) holds. We omit the detail here.

To sum up, Protocol 1 is secure in the semi-honest model and can resist arbitrary collusion attacks. □

5 Protocol Efficiency

In this section, we mainly analyze the efficiency of Protocol 1 and show the experimental results.

5.1 Computational Complexity and Communication Complexity

Computational complexity. In this paper, the protocol is designed under the restriction of the complete set Q with $|Q| = l$. Suppose that the times of each element appearing in the multiset do not exceed m. Protocol 1 is designed based on the elliptic curve cryptosystem, where point multiplication (such as rG) is the most time-consuming operation. Therefore, we measure the computational complexity with the number of point multiplications required for Protocol 1.

Protocol 1 encrypts $ml(n-1)+1$ times and cooperatively decrypts once, therefore, Protocol 1 requires $2ml(n-1)+n+2$ point multiplications. In Protocol 1, the replacement and rerandomization process of $P_2, \cdots, P_{n-1}$ only needs to encrypt zero $ml(n-2)$ times, and these encryption operations can be performed offline. So the online computational complexity of Protocol 1 is $2ml+n+2$ times point multiplications.

Communication complexity. In secure multiparty computation, communication complexity is often measured by the number of communications.

In Protocol 1, the joint generation of public key and joint decryption require n times of communications, and n participants' interaction also require n times of communications. So Protocol 1 requires $3n$ times of communications.

5.2 Experiment

Experimental environment. The configuration of the experimental environment is as follows: Windows10 64-bit operating system, Intel(R) Core(TM) i5-9400 CPU 2.90GHz with 16.0GB memory. The experiments are simulated on PyCharm platform using Python 3.9.4 language in the environment described.

Experimental method. We analyze the execution time of the protocol with the growth of l and n by experiment test. The experimental results are shown in Figure 1 and Figure 2.

It can be seen from Figure 1 that when n is fixed, the execution time increases linearly as l increases. It can be seen from Figure 2 that when l is fixed, the execution time increases linearly as n increases.

6 Conclusions

This paper mainly studies a new problem: privately computing the sum of the elements of the intersection of multisets. First, we propose a new coding method to transform the problem into a matrix computing problem. Then, we use threshold elliptic curve cryptosystem to realize private matrix computing and use the simulation paradigm to prove that our protocol is secure in the semi-honest model. Theoretical analysis and experimental results show that the protocol is efficient. In the future, we will further study the related problems in the malicious model.

References

[1] M. Blanton, E. Aguiar, "Private and oblivious set and multiset operations", *International Journal of Information Security*, vol. 15, no. 4, pp. 493-518, 2016.

[2] J. W. Dou, M. Y. Chen, "Secure computation and application of multiple sets", *Acta Electronica Sinica*, vol. 48, no. 1, pp. 204-208, 2020 (in chinese).

[3] O. Goldreich, *Foundations of Cryptography: Basic Applications.* London: Cambridge University Press, 2004.

[4] S. Goldwasse, "Multi-party computations: past and present", *Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing*, pp. 1-6, New York, USA, 1997.

[5] Y. G. Guan, R. X. Lu, *et al.* "Toward oblivious location-based k-nearest neighbor query in smart cities", *IEEE Internet Things J*, vol. 8, no. 18, pp. 14219-14231, 2021.

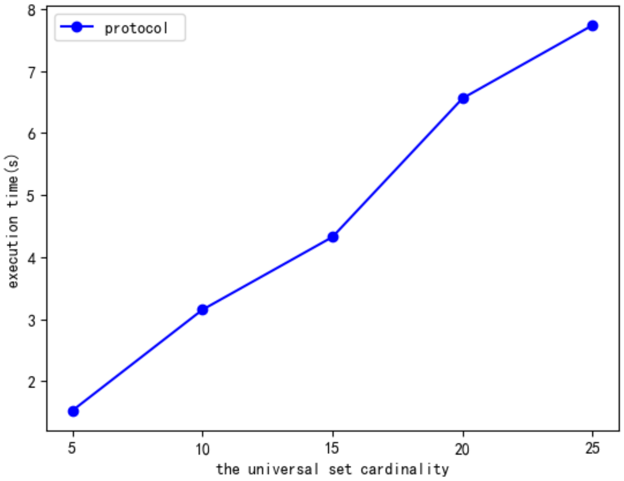


Figure 1: Protocol 1 execution time varies with the universal set cardinality l

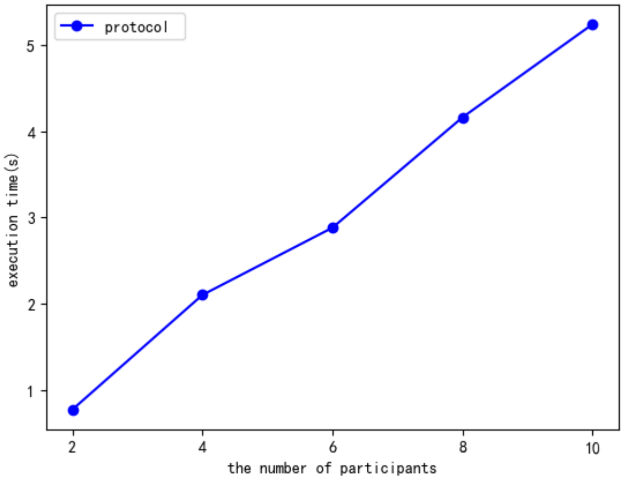


Figure 2: Protocol 1 execution time varies with the number of participants n

[6] D. Hankerson, A. J. Menezes, S. Vanstone, *Guide to elliptic curve cryptography.* Berlin: Springer Science & Business Media, 2006.

[7] L. C. Huang, M. S. Hwang, "Two-party authenticated multiple-key agreement based on elliptic curve discrete logarithm problem", *International Journal of Smart Home*, vol. 7, no. 1, pp. 9-18, 2013.

[8] M. S. Hwang, E. F. Cahyadi, *et al.* "An improvement of the remote authentication scheme for anonymous users using an elliptic curve cryptosystem", *IEEE 4th International Conference on Computer and Communications*, pp. 1872-1877, Chengdu China, 2018.

[9] M. S. Hwang, C. C. Lee, J. Z. Lee, C. C. Yang, "A secure protocol for bluetooth piconets using elliptic curve cryptography", *Telecommunication Systems*, vol. 29, no. 3, pp. 165-180, 2005.

[10] M. S. Hwang, S. F. Tzeng, C. S. Tsai, "Generalization of proxy signature based on elliptic curves", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 73–84, 2004.

[11] D. Josep, R. Sara, *et al.* "Outsourcing scalar products and matrix products on privacy-protected unencrypted data stored in untrusted clouds", *Information Science*, vol. 436, no. 1, pp. 320-342, 2018.

[12] L. Kissner, D. X. Song, "Privacy-preserving set operations", *The 25th Annual International Cryptology Conference*, pp. 241-257, Santa Barbara, California, USA, 2005.

[13] L. H. Liu, Z. J. Cao, "Analysis of a privacy preserving ranked multi-keyword search scheme", *I.J.of Electronics and Information Engineering*, vol. 12, no. 2, pp.76-82, 2022.

[14] L. H. Liu, L. L. Wang, Z. J. Cao, "A note on one protocol for subset sum problem", *I.J. of Electronics and Information Engineering*, vol. 10, no. 2, pp. 98-102, 2019.

[15] C. Melissa, M. Peihan. "Private set intersection in the internet setting from lightweight oblivious PRF", *Proceedings of the 40th Annual International Cryptology Conference*, pp. 34-63, Santa Barbara, CA, USA, 2020.

[16] I. Mihaeal, K. Ben, *et al.* "On deploying secure computing: private intersection-sum-with-cardinality", *IEEE European Symposium on Security and Privacy 2020*, pp. 370-389, Genoa, Italy, 2020.

[17] Y. G. Peng, L. Wang, *et al.* "LS-RQ: A lightweight and forward-secure range query on geographically encrypted data", *IEEE Trans. Dependable Secur. Comput,* vol. 19, no. 1, pp. 388-401, 2022.

[18] M. Peihan, P. Sarvar, *et al.* "Two-sided malicious security for private intersection-sum with cardinality", *Proceedings of the 40th Annual International Cryptology Conference*, pp. 3-33, Santa Barbara, CA, USA, 2020.

[19] O. Tatsuaki, T. Katsuyuki, "Adaptively attribute-hiding (hierarchical) inner product encryption", *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. 99, no. 1, pp. 92-117, 2016.

[20] C. C. Yang, T. Y. Chang, M. S. Hwang, "A new anonymous conference key distribution system based on the elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 25, no. 2, pp. 141-145, 2003.

[21] A. Yao, "Protocols for secure computations", *Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science*, pp.160-164, Chicago, Illinois, USA, 1982.

[22] S. N. Zhang, R. X. Lu, *et al.* "Preserving location privacy for outsourced most-frequent item query in mobile crowdsensing", *IEEE Internet Things J*, vol. 8, no. 11, pp. 9139-9150, 2021.

Biography

Jiahao Pan is currently pursuing the M.S. degree with School of Mathematics and Statistics in Shaanxi Normal University. His research interests focus on cryptography and applied mathematics.

Jiawei Dou received her doctor degree in 2003 at Xi'an Jiaotong University. She is a associate professor of mathematics of Shaanxi Normal University. Her research interests focus on cryptography and applied mathematics.

Data Mining-Based Malicious Traffic Classification Algorithm for Campus Networks

Ziai Wu

(Corresponding author: Ziai Wu)

Chizhou University, Chizhou, Anhui 247000, China

Philippine Christian University Center for International Education, Manila 1006, Philippine

Office of College of Business, Chizhou University, Education Park, Chizhou, Anhui 247000, China

Email: zzi96o@126.com

(Received Mar. 24, 2022; Revised and Accepted Mar. 23, 2023; First Online Apr. 27, 2023)

Abstract

The safe and efficient operation of the campus network is the basis for everyday teaching and learning in schools, so it is of practical significance to classify the malicious traffic in the campus network and make a targeted defense. Based on data mining techniques, this paper constructed a data set, performed feature vector selection, built the Random Forest (RF) algorithm model to classify the traffic, and compared the classification results with those of decision tree (DT) and Support Vector Machine (SVM) algorithms. The experimental results showed that no matter how the percentage of malicious traffic in the total network traffic data changed, the recognition accuracy of the RF algorithm always remained around 96%; at the same time, its missing report rate was 6.67%, the precision was 96.55%, the recall rate was 93.33%, and the F-value was 0.949. The RF algorithm performed better in recognition and classification than the other two classification algorithms. It is concluded that compared with other algorithms, the RF algorithm has better detection speed and classification accuracy and can also classify malicious traffic with high accuracy in the case of a small amount of malicious traffic, which can better prevent security problems in campus networks.

Keywords: Campus Network; Data Mining; Malicious Traffic; Random Forest Algorithm

1 Introduction

Campus network is currently an indispensable part of teaching and learning in colleges and universities, and it is a relatively open area in the whole Internet [17], with many terminals and a high degree of information sharing. Students can obtain course schedules, grades, professional knowledge, and other information from the campus network. However, due to the epidemic in recent years, the emergence of various new types of online teaching software has expanded the openness of the campus network; as a result, more malicious traffic enters the campus network, making the management of the campus network more difficult. Some studies of network malicious traffic are reviewed below.

Fang *et al.* [7] proposed a new method for detecting malicious Transport Layer Security (TLS) traffic based on communication channels. The method selected features from distribution features, consistency features of TLS handshake fields, and statistical features using a genetic algorithm. The experimental results showed that compared with other classification methods, the proposed method had more stable detection efficiency on different data sets, an accuracy of 97.65%, and a higher F1 score.

Chen *et al.* [4] proposed a new network traffic classification model-ArcMargin, validated the proposed model with three data sets, and found that ArcMargin had better performance in both network traffic classification task and open set task. In order to verify the data set collected using DOROTHEA can be used to fit a classification model for malicious traffic detection, Campazas-Vega *et al.* [3] performed experiments and found that all four models constructed with MoEv obtained detection rates higher than 93%.

Yang *et al.* [16] constructed a deep learning-based malicious traffic detection model for encrypted networks and performed automatic feature extraction for encrypted malicious network traffic. They found through experiments that the model could distinguish between normal and abnormal encrypted network traffic, with an accuracy of 99.94%. Al-Fawa'Reh *et al.* [1] studied the likelihood of exposing zero-day malicious network traffic in a large campus network based on cloud environment and found through experiments that it was 100% accurate for specific types of attacks and 97.97% accurate as a comprehensive detection mechanism.

Liu *et al.* [10] constructed a malicious traffic detection model with a hierarchical attention mechanism and proved that the model performed well in terms of different evaluation indicators, including detection rate, false positive rate, and F-score. The purpose of this paper is to detect and classify malicious traffic in campus network environment. This paper built a model based on the random forest (RF) algorithm to classify campus network traffic and carried out experiments after constructing a data set and selecting feature vectors. The classification results of the RF algorithm were compared with those of the Support Vector Machine (SVM) algorithm and the Decision Tree (DT) algorithm to prove the feasibility of the RD algorithm model in classifying malicious traffic. This paper provide a reference for better detection of malicious traffic in campus networks.

2 Campus Network Malicious Traffic Classification

2.1 Campus Network Malicious Traffic

The campus network has many end-users, resulting in a jumble of data traffic on the network, most of which is normal data traffic, but some traffic has attack behavior that can cause great harm. Traffic with attack behavior can be called malicious traffic. Malicious traffic can be divided into four categories: network attacks, account attacks, traffic fraud, and malicious crawlers.

Most of them come from automated programs, usually without permission to invade, interfere with the normal network or crawl network data. For example, ticket scalpers use automated software to grab tickets, some people crack the login to the account and steal the property in the account, companies or individuals hire others to increase page views and followers; Distributed Denial of Service (DDoS) attacks cause the server to stop working properly, etc. The above traffic is all malicious. Some scholars have proposed network traffic-based anomaly detection technology for such malicious traffic [8].

Constructing classifiers to identify network traffic patterns through data mining technology and detecting the presence of abnormal traffic in the network can make campus network maintenance personnel make timely and effective active defense to better maintain the stability and safe operation of campus networks.

2.2 Random Forest Algorithm

Network traffic classification is the process of constructing a classification model using an algorithm to classify traffic with the same characteristics into a class. The traditional network traffic classification methods are based on port, host behavior, or active load [12], but the current emergence of more new types of traffic has led to the low practicality of these three methods. The network traffic classification problem can be divided into supervised, semi-supervised, and unsupervised. According to related studies, the supervised classification method is slightly time-consuming and labor-intensive but relatively accurate, and its applicable algorithms include DT [11], neural network [14], RF algorithms [2].

In this study, the RF algorithm in data mining technique was used to construct a model to classify campus network traffic and find out malicious traffic. The RF algorithm is an integrated learning algorithm based on DT, and its randomness is reflected in the fact that not all the features to be selected are used in the bifurcation process of each of its subtrees, but some features are randomly selected among all the features to be selected, and then the optimal features are selected from these features. Its model building process is as follows.

1) k samples are randomly selected from the initial data set and randomly put back to sampling, which can form k training sets.

2) The k training sets are trained to obtain k DTs.

3) Feature vectors are randomly selected, and the best feature is selected from each DT for bifurcation.

4) The generated k DTs form a RF. The results of all DTs are voted, and the one with the highest votes is the final classification result of the model.

3 Case Analysis

3.1 Data Source and Processing

Since there are few public data sets related to campus network traffic, the author decided to build the experimental initial data set containing normal network traffic and malicious network traffic by himself. The data collection method was to use different probe devices of the campus network for initial screening and collection of campus network traffic. The traffic was selected from July to December 2022. A total of 10,326 traffic data was collected, and some of the traffic data were extracted and converted into malicious traffic by using Flightsim, a network security tool. After conversion, the number of normal traffic was 5,679, and the number of malicious traffic was 4,647.

Moreover, as there were some inconsistent data in the data set, these data were preprocessed to meet the input requirements of the algorithm model. The data processing operations are as follows. First, whether the format of the initial data set was pcap format was checked. If not, the format was changed. The length of each traffic data was unified as 1,024; the excess was intercepted, and the deficiency was supplemented by 0. The missing features and wrong information features were found out through information gain [15] calculation, and logical relations were used to supplement these features. Each feature was calculated and normalized to generate the final feature vector. The processed data set was divided into a training set and a test set for model training and testing, respectively.

3.2 Experimental Design

The campus network traffic data were collected first to build the experimental data set. Then, the data were processed through format unification, length interception, data supplement and correction to achieve the requirements for the input model. Then, the feature importance of the campus network traffic was calculated according to the IG value and Pearson correlation coefficient, and the relatively important features were selected as the input of the RF classification model.

The processed data set was divided into a training set and a test set, and the training set was fed into the RF classification model for feature learning and training. After training, the test set was input into the classification model for classification of malicious traffic to verify the classification effect of the RF classification model. Finally, the RF classification model was compared with SVM [6] and DT models to verify the effectiveness of the RF algorithm model.

3.3 Evaluation Indicators

The ultimate purpose of the experiment is to determine whether the input data is malicious traffic, and therefore it is a binary classification problem. The most commonly used evaluation indicators in the binary classification problem, including the missing report rate, the precision, the recall rate, and the F-value [13], were chosen to evaluate the overall performance of the model. The numerical results were used to test the feasibility of the model. When the values of the precision, recall rate, and F-value were closer to 1, it proved that the model was better at classification.

The missing report rate indicates the percentage of samples that are positive actually but are predicted as negative, and the formula is defined as:

$$FNR = \frac{FN}{TP + FN}$$

The precision rate represents the percentage of samples predicted to be positive among all samples of network traffic, and the formula is defined as:

$$P = \frac{TP}{TP + FP}$$

The recall rate represents the percentage of samples that are positive actually and are predicted as positive, and the formula is defined as follows:

$$R = \frac{TP}{TP + FN}$$

where TP is the number of traffic correctly determined as malicious, TN is the number of traffic correctly determined as normal, FP is the number of traffic incorrectly determined as malicious, and FN is the number of traffic incorrectly determined as normal.

The precision and recall rate were comprehensively assessed by the F-value, and its formula is defined by the following equation:

$$F = \frac{P * R}{P + R} * 2$$

3.4 Analysis of Results

Before starting the experiment the data features of the input RF model were extracted to form a feature vector as input for malicious traffic identification and classification. Ten features, including the number of bytes sent/received, the average session time, the average value of connection period, the five-tuple feature in the packet (source IP, source port, destination IP, destination port, and transport layer protocol), the frequency of request URLs, the server domain name feature, etc., were set, and their information gain values [9] and Pearson correlation coefficients [5] were calculated.

In general, the closer the absolute value of the Pearson correlation coefficient was to 1, the stronger the correlation between two features; the closer the absolute value of the coefficient was to 0, the weaker the correlation between two features. It was seen from the Pearson correlation coefficient values in Table 1 that the absolute values of the coefficients of the ten features were all lower than 0.5, i.e., the features were relatively independent of the other and could be used for the subsequent RF model classification. A larger IG value indicated the more important features. It was seen from the data in Table 1 that the IG value of the five-tuple feature in the data package was the highest, reaching 0.311, which was the only feature with an IG value above 0.3; the coefficient of variation of the request interval and the Gini value of the ID connection had lower IG values compared with the rest of the features, with values hovering at 0.1; the IG values of the other seven features did not differ much and were above 0.2. Therefore, the coefficient of variation of the request interval and the Gini value of the ID connection were removed, and the remaining eight features with high importance were selected as the feature vectors for the input of the RF algorithm.

Figure 1 shows the line graphs of the precision of RF and SVM algorithms under different percentages of malicious traffic in the total network traffic. It was seen from Figure 1 that the accuracy precision of SVM and DT algorithms fluctuated greatly. The precision was the highest, 95.94% and 92.91%, respectively, when the malicious traffic accounted for 45% of the total traffic. When the percentage was 25% and 35%, the precision increased synchronously as the percentage increased. When the percentage was 55% and 65%, the precision decreased, although the percentage kept increasing. Different percentages represented the different balance of the data set; the more the percentage converged to 0.5, the higher the balance of the data set.

Therefore, it was concluded that the balance of the data set had some influence on the classification perfor-

Table 1: Selection of features of malicious traffic

Feature name	IG value	Pearson correlation coefficient
Number of bytes sent/received	0.294	0.4201
Average session duration	0.261	0.3926
Average value of connection period	0.249	0.1923
Five-tuple feature in the packet	0.311	0.2463
Coefficient of variation of the request interval	0.103	0.2137
Frequency of request URLs	0.251	0.3447
Gini value of ID connection	0.117	0.1954
TLS handshake feature	0.243	0.3657
Server certificate feature	0.257	0.2689
Server domain name feature	0.289	0.3462

mance of SVM and DT models, and it was necessary to keep the balance of the data set as much as possible to get the best classification performance. However, Figure 1 shows that the broken line of the RF algorithm was relatively stable, but its precision always maintained around 96%, and there was no significant upward or downward trend with the increase of the proportion of malicious traffic. It was seen that for the unbalanced data set, the RF algorithm could balance the error, suggesting good recognition and classification performance.

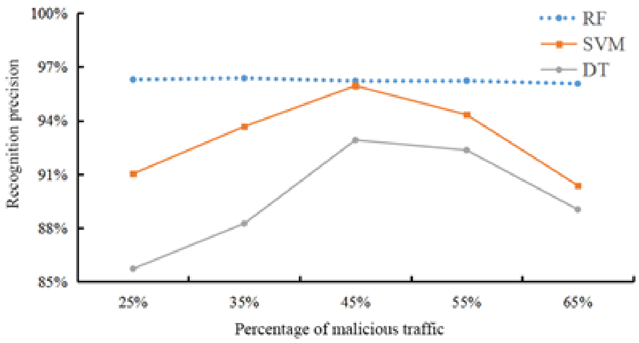


Figure 1: Experimental results of malicious traffic at different percentages

At the end of the experiment, the classification results of the RF classification algorithm were compared with those of the SVM and DT classification algorithms. In terms of computing time, the RF algorithm differed little from the DT algorithm, but it was slightly higher than that of the DT algorithm and was 2.19 s faster than that of the SVM algorithm. In terms of the missing report rate, the RF algorithm was only 6.67%, which was the lowest among the three classification algorithms and was much lower than the other two classification algorithms. In terms of precision, recall rate, and F-value, these values of the RF algorithm were higher than those of the SVM and DT algorithms, reaching 96.55%, 93.33%, and 0.9492, respectively. According to the above data, the RF algorithm had the best classification performance, followed by the SVM algorithm and the DT algorithm. These results proved that the RF algorithm could well classify the malicious traffic present in the campus network accurately.

4 Conclusion

This paper briefly introduced the malicious traffic and the RF classification algorithm and studied the classification of malicious traffic in campus network based on the RF algorithm in data mining technology. Before the experiment, the author constructed a data set by himself and carried out data pre-processing. The data were divided. The data in the training set were used for feature learning and training of the RF model, and the test set was input to the optimal model to obtain the final experimental results. The results showed that no matter how the percentage of malicious traffic in the total data changed, the recognition accuracy of the RF algorithm was maintained around 96%, which was relatively stable; at the same time, its missing report rate, precision, recall rate, and F-value were 6.67%, 96.55%, 93.33%, and 0.9492, respectively, which were better than the two classification algorithms, SVM and DT. These results prove that the RF algorithm can still maintain a stable and high recognition rate for unbalanced data and has high efficiency and accuracy in classifying malicious traffic in campus network. The RF algorithm can guarantee the normal and safe operation of campus network and provide a solid foundation for the teaching work of the school.

References

[1] M. Al-Fawa'Reh, M. A. Al-Fayoumi, "Detecting stealth-based attacks in large campus networks," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 4, pp. 4262-4277, 2020.

[2] S. Bagui, J. Simonds, R. Plenkers, T. A. Bennett, S. Bagui, "Classifying UNSW-NB15 network traf-

Table 2: Experimental results of different classification algorithms

	Computing time	Missing report rate	Precision	Recall rate	F-value
The RF algorithm	17.28s	6.67%	96.55%	93.33%	0.9492
The SVM algorithm	19.47s	16.67%	89.29%	83.33%	0.8621
The DT algorithm	17.52s	30.00%	72.41%	70.00%	0.7119

fic in the big data framework using random forest in spark," *International Journal of Big Data Intelligence and Applications*, vol. 2, no. 1, pp. 39-61, 2021.

[3] A. Campazas-Vega, I. S. Crespo-Martínez, Á. M. Guerrero-Higueras, C. Fernández, "Flow-data gathering using NetFlow sensors for fitting malicious-traffic detection models," *Sensors*, vol. 20, no. 24, pp. 1-13, 2020.

[4] M. Chen, X. Wang, M. He, L. Jin, K. Javeed, X. Wang, "A Network Traffic Classification Model Based on Metric Learning," *Computers, Materials & Continua*, vol. 64, no. 2, pp. 941-959, 2020.

[5] P. Chen, F. Li, C. Wu, "Research on intrusion detection method based on pearson correlation coefficient feature selection algorithm," *Journal of Physics: Conference Series*, vol. 1757, no. 1, pp. 1-10, 2021.

[6] S. Dong, "Multi class SVM algorithm with active learning for network traffic classification," *Expert Systems with Applications*, vol. 176, 2021.

[7] Y. Fang, K. Li, R. Zheng, S. Liao, Y. Wang, "A communication-channel-based method for detecting deeply camouflaged malicious traffic," *Computer Networks*, vol. 197, pp. 1-14, 2021.

[8] Y. Feng, W. Cai, H. Yue, J. Xu, Y. Lin, J. Chen, Z. Hu, "An improved X-means and isolation forest based methodology for network traffic anomaly detection," *PLoS ONE*, vol. 17, no. 1, pp. 1-18, 2022.

[9] T. Gao, T. Li, R. Jiang, M. Yang, R. Zhu, "Research on cloud service security measurement based on information entropy," *International Journal of Network Security*, vol. 21, no. 6, pp. 1003-1013, 2019.

[10] X. Liu, J. Liu, "Malicious traffic detection combined deep neural network with hierarchical attention mechanism," *Scientific Reports*, vol. 11, no. 1, pp. 1-15, 2021.

[11] P. Nancy, S. Muthurajkumar, S. Ganapathy, S. V. N. Santhosh Kumar, M. Selvi, K. Arputharaj, "Intrusion detection using dynamic feature selection and fuzzy temporal decision tree classification for wireless sensor networks," *IET Communications*, vol. 14, no. 5, pp. 888-895, 2020.

[12] J. Oluranti, N. Omoregbe, S. Misra, "Effect of feature selection on performance of internet traffic classification on NIMS multi-class dataset," *Journal of Physics: Conference Series*, vol. 1299, pp. 1-10, 2019.

[13] S. Phetlasy, S. Ohzahata, C. Wu, T. Kato, "A sequential classifiers combination method to reduce false negative for intrusion detection system," *IEICE Transactions on Information and Systems*, vol. E102.D, no. 5, pp. 888-897, 2019.

[14] O. Salman, I. H. Elhajj, A. Kayssi, A. Chehab, "Data representation for CNN based internet traffic classification: a comparative study," *Multimedia Tools and Applications*, vol. 80, no. 11, pp. 16951-16977, 2021.

[15] F. Salo, A. B. Nassif, A. Essex, "Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection," *Computer Networks*, vol. 148, no. JAN.15, pp. 164-175, 2019.

[16] J. Yang, G. Liang, B. Li, G. Wen, T. Gao, "A deep-learning- and reinforcement-learning-based system for encrypted network malicious traffic detection," *Electronics Letters*, vol. 57, no. 9, pp. 363-365, 2021.

[17] W. Zhang, Z. Qin, Z. Feng, J. Liu, W. Liu, X. Tang, "Big data analysis for detection of web brute-force attack," *Journal of Shenzhen University Science and Engineering*, vol. 37, no. Z1, pp. 44-49, 2020.

Biography

Ziai Wu, male, born in Yujiang, Jiangxi Province, in 1981, is a Ph.D. candidate, an associate professor, and the vice dean of Business School, Chizhou University. His research interests include e-commerce and information technology, teaching reform of higher education.

Mining Method of Code Vulnerability of Multi-Source Power IoT Terminal Based on Reinforcement Learning

Hao Yang[1], Junfeng Zhang[2], Jun Li[2], and Xin Xie[3]
(Corresponding author: Xin Xie)

State Grid Jiangxi Electric Power Research Institute, Nanchang, China[1]
State Grid Jiangxi Electric Power Co., Ltd, Nanchang, China[2]
East China Jiaotong University, Nanchang, China[3]
Email: xiexin@ecjtu.edu.cn

(Received Oct. 17, 2022; Revised and Accepted Mar. 13, 2023; First Online Apr. 27, 2023)

Abstract

With the rapid development of IoT, many heterogeneous power terminals are connected, substantially increasing the difficulty of network attack protection. How to accurately grasp the supporting techniques such as sample generation, vulnerability targeting, and vulnerability correlation in intelligent vulnerability mining to improve the efficiency of vulnerability mining and ensure grid security is a major challenge we are currently facing. This paper studies the multi-source power IoT terminal code vulnerability mining method based on reinforcement learning. Firstly, the static analysis method is used to scan and analyze the source code of the multi-source power IoT terminal, and the abstract syntax tree of the code is constructed. A bidirectional search path algorithm is adopted to expand the path range of the directed graph of the multi-source power Internet of things terminal. Secondly, the vulnerability of multi-source power IoT terminal code is located by the concept of dynamic taint tracking. The taint tracking results are input into the deep neural network model as samples. Finally, the protocol vulnerability mining model based on reinforcement learning is constructed to obtain the vulnerability mining results. The experimental results show that the method has high vulnerability mining accuracy and coverage, can record the type and location of vulnerabilities, and generate vulnerability reports to improve the security of the smart grid.

Keywords: Dynamic Stain Tracking; Path Algorithm; Power IoT Terminal; Reinforcement Learning; Vulnerability Mining

1 Introduction

Emerging intelligent devices are rapidly promoting the popularization of intelligent life, and as an important part of the Energy Internet, which play a vital role in the intelligent and automatic production of the power grid, such as data terminal units, remote terminal units, feeder terminal units, smart meters, relay protection devices, etc. [6]. The intelligent terminal of the power grid influences the power production process through monitoring [25], control and protection, such as remote terminal units can influence power production by opening and closing the power line, and monitoring the voltage and current to protect and control the power production in real-time [22]. With the rapid development of the Internet of things (IoT), the security problems behind IoT have become increasingly prominent [13,15]. These networked smart devices can make people's life more convenient, but security vulnerabilities may bring great potential harm to power grid companies and users [4].

Taking data terminal units as an example, hackers are very good at attacking the intelligent terminal detection equipment by using the hidden back door vulnerability [23] [24]. Because the back door is a vulnerability intentionally integrated in the embedded device and can provide remote access to anyone with "secret" authentication information, the malicious attacker will conduct malicious operations and steal sensitive information through the back door vulnerability [20]. Smart meters are also one of the equipment types with serious vulnerabilities. Attackers can use vulnerabilities such as buffer overflow to crash the target equipment, resulting in line overload or even causing a fire in serious cases. There are many serious security vulnerabilities in ZigBee and GSM communication standards used by smart meters. However, when manufacturers use GSM networks, many power devices still do not introduce any form of encryption, and attackers can hijack communication data and gain control of target devices [9,12]. There are more or less code security vulnerabilities, software package vulnerabilities, sensitive information leakage and other security risks in terminal devices, which may be maliciously attacked by attackers, resulting in the execution of malicious operations, information theft or equipment paralysis.

By analyzing security incidents of IoT devices, it can be found that there are certain rules in the chain of attacks on IoT devices [2]. For example, the attacker can analyze the operation process and network behavior of the device after obtaining and unpacking the firmware of the device. It can also find the key information related to security encryption, to carry out targeted vulnerability attacks. From this point of view, the security of IoT devices depends largely on the security of their firmware [18].

With the development of the smart grid, a large number of diversified and heterogeneous power terminals are connected to the power grid. These power terminals not only provide various functions and convenience to the power grid, but also bring more security threats [10]. Unlike the traditional power grid, there is a two-way flow of information and data between smart grids. The original "isolation" protection cannot effectively prevent more attacks when a large number of power terminals are connected. The power terminal itself has a large number of software vulnerabilities, and most of them are embedded terminal devices, which have the characteristics of slow update. Once the vulnerability exists, it may be latent for a long time and cannot be repaired in time. This allows the attacker to seize control by attacking these power terminal devices, and further use this as a springboard to launch more attacks on the master station or control center [21]. Therefore, ensuring the information security of power terminals is the premise of ensuring the safe and stable operation of the smart grid [28], so many scholars study the methods of vulnerability mining.

Lai *et al.* [11] proposed a Modbus TCP vulnerability mining test case generation model based on the anti-sample algorithm. First, the recurrent neural network is trained to learn the semantics of the protocol data unit. The probability distribution of the data is expressed by the softmax function. Then the random variable threshold and the maximum probability are compared to determine whether to replace the current data with the minimum probability data. Finally, the Modbus application protocol (map) header is completed according to the protocol specification. The method not only improves the acceptance rate and vulnerability utilization ability but also detects the vulnerabilities more quickly. However, the false positive rate is too high, which will bring a lot of unnecessary workloads. Chu *et al.* [5] solved challenges to network security caused by botnet detection, vulnerability mining and confrontation, which propose a botnet vulnerability mining and countermeasure algorithm. First, a botnet model based on machine learning is designed. Then, to realize global optimal evaluation and screening of botnet detection, a combination of classification mining algorithms and machine learning instruction sets are designed. The method can quickly mine network vulnerabilities and ensure network security. However, it can only improve the mining rate, the accuracy rate of vulnerability mining is relatively low, which cannot achieve the effect of vulnerability detection. Men *et al.* [16] have studied the discovery of IoT vulnerabilities in the human-machine interface. They start with code classification and qualitative description of code features to explore the idea of code similarity and homology analysis, which focuses on the information of the firmware code gene, and through discussion and analysis of the similarity and homology of the firmware code, it provides a basis for mining the vulnerabilities of the IoT. The method can quickly find similar vulnerabilities through the similarity and homology of the firmware code vulnerabilities. However, the scope of IoT vulnerability mining through the human-machine interface is small, and only a part of known vulnerabilities can be mined.

Reinforcement learning is one of the methodologies of machine learning, which is used to describe and solve the problem that an agent uses learning strategies to maximize returns or achieve specific goals in the process of interacting with the environment [1]. Therefore, a mining method based on reinforcement learning target at code vulnerability of multi-source power IoT terminal is proposed to expand the scope of vulnerability mining and the correlation between vulnerabilities, improve the accuracy of mining, and ensure the code vulnerabilities be quickly and accurately mined in the case of multi-source data accessing the IoT. The main contribution of this method is:

1) Common support technologies such as sample generation, vulnerability orientation and vulnerability association are proposed to provide a foundation for the vulnerability mining of multi-source power IoT terminal code.

2) Use the dynamic stain tracking concept to intelligently mine the vulnerabilities of the multi-source power IoT terminal code.

3) The taint tracking results are input into the deep neural network model as samples to analyze and verify the security of the protocol and detect the security of multi-source power IoT terminal.

4) Experimental results show that the method has good accuracy and a false alarm rate. Most of the vulnerabilities can be detected without a high false alarm rate.

2 Method

Aiming at the problems of weak ability and low efficiency of traditional vulnerability mining technology, the research and verification of intelligent vulnerability mining technology of power IoT terminals are carried out. Common support technologies such as sample generation, vulnerability orientation and vulnerability association will provide support for intelligent vulnerability mining of power IoT terminals. In addition, the dynamic stain analysis technology is improved to realize the intelligent mining of firmware vulnerabilities. This paper analyzes and verifies the protocol security based on the deep learning algorithm, establishes a protocol vulnerability mining model based on reinforcement learning, and improves the accuracy of vulnerability mining and test coverage.

2.1 Common Support Technology for Intelligent Vulnerability Mining

2.1.1 Vulnerability Sample Generation Based on Static Analysis

Program static analysis technology is often used in mainstream vulnerability mining methods, which analyze the program code without executing it, which is usually used to scan the code and extract the program features for subsequent analysis [27] [7]. In this paper, the common lexical and syntax analysis methods in the static analysis are used to scan and analyze the source code of the multi-source power IoT terminal, and construct the abstract syntax tree of the code. Meanwhile, symbol execution and model checking are adopted to check the abstract syntax tree model of the code and match the vulnerability pattern [17].

The specific process about sample generation of the source code of multi-source power IoT terminal is to carry out comprehensive static analysis and feature extraction of the code, including lexical analysis and syntax analysis, and build an abstract syntax tree of the code, which is taken as the sample of the code, as shown in Figure 1.

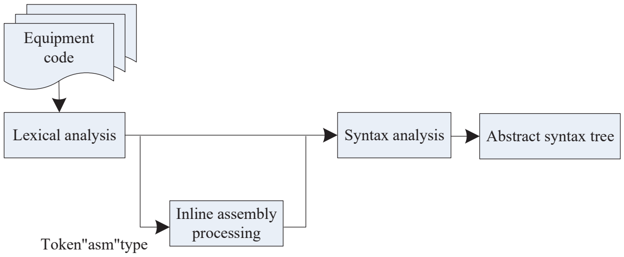


Figure 1: Sample generation process of multi-source power iot terminal code

Specifically, the implementation of static analysis borrows the idea of the compiler front-end. Figure 2 shows the implementation process of a compiler. The compiler front-end generates intermediate code after preprocessing, lexical analysis, syntax analysis and semantic analysis of C source code, and then the intermediate code is optimized and processed by the compiler back-end to generate executable binary code [3]. The tool implemented in this paper mainly needs to carry out a static analysis of the source code. Referring to the idea of the compiler front-end, the method of lexical analysis and syntax analysis can be used. Finally, the abstract syntax tree model of the source code can be constructed. At the same time, specific rules can be written to check out some nonstandard code writing and syntax errors.

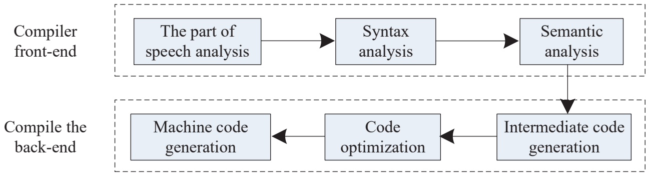


Figure 2: C source code compilation process

PLY (Python Lex yacc) is the python implementation of lex and yacc. With the help of PLY library, we can construct syntax rules to realize the static analysis of a specific language. The main process of the static analyzer is as follows:

1) Use lex to construct a proper lexical analyzer, and convert the input source code of multi-source power IoT terminal into a token sequence. In particular, we define a token with the keyword “ASM” to identify the inline assembly in the code for subsequent processing.

2) Preprocess the token sequence, remove the spaces and comments in the code, merge the multiple lines of code, and process the conditional compilation instructions, macro definitions and header files. Macro substitution is required when dealing with macro definitions. When dealing with header file inclusion, the above processing also needs to be performed recursively for the included header file.

3) In the token sequence preprocessing, if a token with “ASM” is encountered, the inline assembly processing module is called to process the relevant sequence. The module recognizes the assembly instructions in the “ASM{}” format, recognizes and processes the specific instructions, and generates the C language expression.

4) The preprocessed token sequence does not contain spaces and comments, preprocessing instructions starting with #, nor inline assembly, but only contains the actual code token.

5) The yacc module is used to construct an appropriate parser, and the C language grammar is defined according to the C99 standard to specify the syntax rules. The preprocessed token sequence is parsed and the syntax structure is recognized to generate an abstract syntax tree, which is used as a sample of the multi-source power IoT terminal code.

2.1.2 Vulnerability Orientation

The vulnerability orientation of multi-source power IoT terminal code adopts a two-way search path algorithm to expand the mining coverage of vulnerabilities [14]. Path generation based on the two-way search is to automatically obtain the path from one code location to another. It first obtains the control flow diagram, function call diagram and key code area of binary code by static analysis method [19]. The two-way path search technology is adopted to automatically obtain all program paths from the input location of the binary terminal to its code area from the input location of the tested terminal and the key code area [8].

The control flow of the binary code can be represented by the directed graph $Q = \langle V, O\rangle$, where $V = \{v_1, v_2, \cdots, v_n\}$ represents the nodes of the directed graph, n represents the number of nodes, $O = \{e_1, e_2, \cdots, e_m\}$ represents the edges in the directed graph, and M represents the number of edges. The incidence matrix of the directed graph Q of the code can be defined as:

$$M(Q) = \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_n \end{pmatrix} = (m_{ij})_{n*m}, \tag{1}$$

$$where\ a_{ij} = \begin{cases} 1 & Node\ v_i\ has\ a\ jump \\ & relationship\ with\ node\ v_j \\ 0 & others \end{cases}$$

$$i = 1, 2, \cdots, n; j = 1, 2, \cdots, m$$

The adjacency matrix of Q can be defined as:

$$A(Q) = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_n \end{pmatrix} = (a_{ij})_{n*n}, \tag{2}$$

$$where\ a_{ij} = \begin{cases} 1 & Node\ v_i\ has\ a\ jump \\ & relationship\ with\ node\ v_j \\ 0 & others \end{cases}$$

$$i = j = 1, 2, \cdots, n$$

The starting point of the binary program control flow diagram is node 1, and the ending point is node 10. The forward search is started from the starting point 1, and the reverse search is started from node 10. The specific steps are as follows:

1) Starting from the starting node of the diagram, traverse its edge as the output point to obtain the first edge set $S_1 = \{(1)(2)\}$ of the forward search.

2) Starting from the end node of the multi-source power IOT terminal code, first judge whether there is an edge with the node 10 as the outgoing point (look up the element that is not 0 in the 10th column of the adjacency matrix A to obtain the elements a_{5*10} and a_{9*10} , then there is an edge from the node 10 to the node 5 and the node 9, and then go through the edge with the node 10 as the outgoing point to obtain the first edge set $T_1 = \{(10)(9), (10)(5)\}$ of the reverse search.

3) Judging whether the end points of the edges in the S_1 obtained by the forward search and the end points of the edges in the T_1 obtained by the reverse search are the same, and the end points of the edges in the edge sets S_1 and T_1 are not the same.

4) Take the end point of each path in the edge set S_1 as the starting point, traverse its edge as the exit point, and obtain the second edge set $S_2 = \{(1)(2)(3), (1)(2)(4)\}$ of the forward search.

5) Taking the end point of each path in T_1 as the starting point, first judge whether there is an edge with node 9 and node 5 as the degree point (look for elements that are not 0 in the 9th and 5th columns of A to obtain elements a_{7*9}, a_{8*9} and a_{4*5}, then there is an edge from node 9 to node 7 and node 8, and an edge from node 5 to node 4), and then traverse the edge with node 9 and node 5 as the degree point respectively, The second edge set $T_2 = \{(10)(9)(7), (10)(9)(8), (10)(5)(4)\}$ (5)of the reverse search is obtained.

6) Judging whether the end points of each path in S_2 obtained by the forward search and the end points of each path in the edge sets T_1 obtained by the reverse search are the same. If there is the same, a path from the start point 1 to the end point 10 is found, and the end points of the paths (1) (2) (4) in S_2 and (10) (5) (4) in T_2 are the same, then a path (1) (2) (4) (5) (10) from the start node 1 to the end node 10 is found, and the path is added to the path set $PathSet = \{(1)(2)(4)(5)(10)\}$, and the path (10) (5) (4) in T_2 is deleted.

7) Repeat the above steps until the edge set of the digraph of the code obtained by the reverse search is the empty set T_n.The path set is the set of paths from the start node 1 to the end node 10.

The basic block where the binary program input position of the multi-source power IOT terminal code is located is the starting point s of the digraph Q. The basic block where the key code area in the binary program of the multi-source power IOT terminal code is located is the end point t of the digraph Q. Starting from the start point s and the end point t, the algorithm of two-way search can obtain all program paths from the start point to the end point, and expand the scope of multi-source power IOT terminal code vulnerability mining [26].

2.1.3 Vulnerability Association

Vulnerability association technology refers to the use of known firmware vulnerabilities to detect homologous vulnerabilities in other firmware, so as to facilitate the subsequent multi-source power IOT terminal code vulnerability mining and improve the efficiency of vulnerability mining.

1) Extract the numerical characteristics of the function

Divide the binary file of the code into multiple functions, and then extract the features of the functions. The specific extraction process is as follows:

a. Function call graph
Function call graph is a directed graph. Nodes represent functions, and directed edges represent the calling relationship between functions. The function call graph of the code is analyzed by the following parameters: callt, the number of times that the function is called by other functions; callf, the number of times that the function calls other functions; callt2, the number of times that the function calls other functions, and call F2 after de duplication are extracted to form the call relationship feature.

b. Function basic properties
Analyze the basic information of the code function, calculate the stack space **stack**, the code amount **code**, the number of call strings **STR**, and the called string set strset. Perform instruction analysis on the function, count the number of instructions **Inst**, the number of jump instructions **jump**, and the proportion of jump instructions **jumpp**, and calculate the instruction entropy **instept** and jump instruction entropy **jumpept** in combination with Equation (3). P_k represents the proportion of (jump) instructions, and K is the total number of instructions. The above characteristics constitute the basic characteristics of the function.

$$Entropy = -\sum_{k=1}^{K} P_k lb P_k. \tag{3}$$

c. Functional control flow graph
One function corresponds to one function control flow graph (CFG). Each node in the graph corresponds to one basic block in the function. The directed edge between nodes corresponds to the jump relationship between basic blocks. The CFG of the source code is analyzed from three aspects: point edge, degree and path.

The CFG of the code function is analyzed from point to edge. Calculate the number of nodes and edges of the CFG; Calculate the density of the graph according to formula (4), which constitutes the attribute characteristics of the points and edges of the CFG.

$$density = \frac{edge \times 2}{node\,(node-1)}. \tag{4}$$

Perform path analysis on the CFG of the code function, calculate the minimum distance from the entry basic block to any basic block by using Floyd or Dijkstra algorithm, construct the ascending distance sequence **pathlist**, calculate the average path length **avepath** and the graph diameter (i.e. the longest path) diameter of the graph, and calculate the graph link efficiency effect according to Equation (5), which constitutes the path characteristics of the CFG.

$$effect = (edge - avePath)\,/edge. \tag{5}$$

Carry out degree analysis on the CFG, count the out degree and in degree of each node, convert the CFG into an undirected graph, calculate the degree of each node of the undirected CFG, and form the in degree ascending sequence **IList**, out degree ascending sequence **olist** and undirected degree ascending sequence **ulist**. From the three degree sequences, respectively calculate three maximum degrees **IMAX**, **OMAX**, **UMAX** and three average degrees **Ieva**, **oeva** and **ueva**; The entropy **uept** of the undirected graph degree is calculated from Equation (1). Calculate the clustering coefficient cluster of the graph according to Equations (6) and (7), where C represents the number of subgraph edges of the undirected CFG composed of all neighbor nodes of node K, d_k represents the degree of node K, and the above constitutes the degree feature of CGF.

$$C_{ck} = \frac{2c}{(d_k-1)\times d_k}. \tag{6}$$

$$cluster = \sum_{k=1}^{K} (C_{ck}/K)\,. \tag{7}$$

The above 31-dimensional features are taken as numerical features of one function. Feature selection is the process of selecting some of the most effective features from the original features to reduce the dimension of the data set. It is an important means to improve the performance of the learning algorithm.

2) Calculate the similarity vector of the function to be matched
Since the problem is the similarity measurement between code functions, not the classification problem, the input of the neural network is not the 31 dimensional feature of one function, but the similarity vector composed of the similarity of each dimensional feature the code functions in sequence type and set type, this paper adopts different similarity measurement methods:

a. For sequence type features: pathlist, IList, olist, ulist. Formula (8) calculate the longest common subsequence (LCS) ratio of the sequence type features L_1 and L_2 of the code functions f and g to be compared as the similarity:

$$sim = \begin{cases} c_0, L_1 = 0 \text{ or } L_2 = 0 \\ \frac{LCS(L_1,L_2)}{\max(L_1,L_2)}, \text{others} \end{cases} \tag{8}$$

Where c_0 is a constant between 0 and 1.

b. For collective features: calculate the Jaccard coefficients of collective features s_f and s_g of code functions f and g as the similarity:

$$sim = \begin{cases} c_1, |s_f| = 0 \text{ and } |s_g| = 0 \\ \frac{s_f \cap s_g}{s_g \cup s_f}, \text{others} \end{cases} \tag{9}$$

Where c_1 is a constant between 0 and 1.

c. For 26 dimensional features and quantitative features. Use formula (10) to calculate the similarity between the quantitative features F_f and F_g of the code functions f and g:

$$sim = \begin{cases} c_2, F_g = 0 \text{ and } F_f = 0 \\ 1, 0 - \frac{|F_f - F_g|}{\max(F_f, F_g)}, \text{others} \end{cases} \tag{10}$$

Where c_2 is a constant between 0 and 1.

Thus, the similarity degree of the 31-dimensional features is obtained, and the value range is [0,1], which constitutes the similarity feature vector of the function pair $(f, g)$of each code to be compared. The higher the similarity of the code function, the stronger the correlation of the code vulnerability.

2.2 Vulnerability Tracking Based on Dynamic Stain

According to the concept of dynamic stain tracking, the possible location of code vulnerabilities of multi-source power IoT terminal is located to facilitate the implementation of subsequent vulnerability mining.

The process of tracking code with the stain tracking algorithm is cyclic. Analyze each target instruction and judge whether it is a stain source. The stain source is the possible location of the code vulnerability. The detailed steps are explained as follows:

Step 1: Analyze the instruction at the current running position to determine its type, instruction function, instruction source operand and instruction destination operand, then proceeds to Step 2.

Step 2: Judge whether the instruction has data write function according to the function and type obtained in Step 1. If not, step forward and return to Step 2. Otherwise, proceed to Step 3.

Step 3: If the instruction has a data writing function and is not a jump instruction, fetch the operand and judge whether the destination operand has the propagation of tainted data, that is, judge whether the operand is related to the tainted source according to the definition of the state model. If yes, set the variable to the polluted state, and then go to Step 5. Otherwise, step proceeds to Step 2.

Step 4: If the instruction is a jump instruction, judge whether the jump parameter is related to the pollution data. If any parameter comes from the stain variable, proceed step by step, and then proceed to Step 1. Otherwise, wait for the return, and then go to Step 1.

Step 5: Judge the instruction from step 3 according to the security rules defined by the attack surface. If yes, the variable associated with the operand is placed into the dangerous state variable set, and the stain source field is obtained. The danger weight associated with the field increases automatically. Finally, the single step advances to Step 1.

2.3 Vulnerability Mining Based on Reinforcement Learning

2.3.1 Deep Learning Network Model for Protocol Security Analysis and Verification

The security of source code of multi-source power IoT terminal is analyzed and verified based on the deep learning algorithm. Generating adversarial network (GAN) is a deep learning and unsupervised learning method. The generation model randomly samples the multi-source power IoT terminal code to form an input. By judging that the input source of the model is from the actual sample of the code or the output of the generation model, the real source of this input can be determined.

The confrontation between the generation of multi-source power IoT terminal code sample model and the discrimination model can be described as:

$$\begin{aligned} \underset{G}{min}\ \underset{D}{max}\ R(D,G) = E_{z \sim Pz(z)} \left[\log\left(1 - D\left(G\left(z\right)\right)\right)\right] \\ + E_{x \sim Pdata(x)} \left[\log D\left(x\right)\right]. \end{aligned} \tag{11}$$

Where P_{data} is the distribution of actual data, $D(x)$ is the output of the discrimination model, P_z is the distribution of generated data, $G(x)$ is the output of the generation model.

After optimization D:

$$\begin{aligned} \underset{D}{max}\ R(D,G) = E_{z \sim Pz(z)} \left[\log\left(1 - D\left(G\left(z\right)\right)\right)\right] \\ + E_{x \sim Pdata(x)} \left[\log D\left(x\right)\right]. \end{aligned} \tag{12}$$

After optimizing:

$$\underset{G}{min}\ R(D,G) = E_{z \sim Pz(z)} \left[\log\left(1 - D\left(G\left(z\right)\right)\right)\right]. \tag{13}$$

The feedforward neural network (FNN) is regarded as the generation model, the support vector machine (SVM) is the discrimination model, the FNN is always straight forward, and the information is transmitted from front to back.

SVM is a learning method used in classification and regression analysis which is a linear classifier. Define its straight line and optimization function respectively:

$$y\left(x\right) = b + w^2 x. \tag{14}$$

$$min\frac{1}{2}\left\|w\right\|_2^2 \quad s.t.y_p\left(b+w^2x_p\right)\geq 1. \tag{15}$$

According to Lagrangian duality, the problem is transformed into an extreme value problem. First, the minimum value of ω and B is solved, and then the maximum value of α is calculated. Finally, the value of ω and B is the protocol vulnerability detection result:

$$w=\sum_{p=1}^{n}a_px_py_p. \tag{16}$$

$$b=y_p-\sum_{p=1}^{n}a_px_py_py_q \quad a_q>0. \tag{17}$$

The generated code samples are combined into a training set, and the training set is used to train and generate an adversarial network. The generated protocol message is input into the protocol system to check the system operation. If the system is abnormal, it indicates that there is a security vulnerability, and the vulnerability location can be accurately found by using Equations (14) and (15).

2.3.2 Vulnerability Mining Model Based on Deep Reinforcement Learning

In order to more efficiently use the dynamic strategy to formally verify the security protocol of the multi-source power IoT terminal, and make the deep learning strategy generalized, it is necessary to optimize the deep learning model in the verification framework. There are two challenges: 1) how to convert the formal verification data into the input of the neural network on the premise that the data information is relatively complete; 2) how to improve and optimize the structure of the deep learning network so that the model can be generalized.

Here, reinforcement learning is introduced to optimize the design of deep learning network. The specific steps are as follows:

Step 1: Select the protocols used in the deep learning protocol security analysis model for random segmentation, take 20 of them as the training set and 4 as the verification set.

Step 2: Train the protocols in the training set in a multi-threaded manner. For each protocol in each round of training:

1) Tamarin prover is used to generate formal data, and the data is converted into a proof theorem tree.
2) The eigenvector is constructed for each node in the current proof theorem tree that has not yet been proved, and the corresponding action selection probability of all these nodes and the value of the current node are calculated using the deep neural network (DNN).
3) Monte Carlo tree search (MCTS) is used for all nodes calculated in step 2 (2), and each node is selected, expanded and updated each time. After 100 repetitions, the action selection of the current node is performed using the results of MCTS, until the action selection is completed for all nodes.
4) Add all the action selections in step 2 (3) to the proof theorem tree, and use the loop detection algorithm to detect each non-terminated proof path in the proof theorem tree.
5) If the path has generated a loop, the path will be marked as a loop path, and no proof will be performed. Otherwise, continue to repeat steps 2 (1) to 2 (4) for the path.
6) If all paths on the proof theorem tree have terminated the proof, it indicates that the current protocol is proved successfully or cannot terminate the proof. It is necessary to evaluate each node on the proof theorem tree and set rewards, and save the data to the data buffer.
7) After the verification of all protocols in this round, 256 data are randomly extracted from the data buffer for training.

Step 3: The DNN is evaluated once every 5 times of training. Compare the best model of the trained and saved DNN, leaving the better model as the new DNN, and taking it to verify the security of the protocol.

The security protocol of deep learning network is trained by formal data generated in the process of reinforcement learning verification. For each feature vector, MCTS is performed under the guidance of the upper deep learning network. The probability of all current optional actions will be generated after the MCTS is completed. The probability is often closer to the desired result (protocol completion verification) than the action probability generated by the deep learning network after MCTS. Therefore, in the process of protocol verification, the MCTS-based enhancement strategy is used to select each action, and then the data that can complete the verification and the data that cannot terminate the verification are used as samples to train the deep learning network. The MCTS process can be regarded as a powerful strategy evaluator. The iterative process of reinforcement learning repetition strategy uses the action selection strategy generated by MCTS as the guidance of the action selection strategy to update the parameters of the deep learning network. The deep learning network parameter update makes the strategy and value generated in the iterative process gradually approach the strategy and value that can successfully verify the protocol, so as to obtain more accurate code vulnerability mining results of the power IoT terminal.

2.4 Implementation of Protocol Vulnerability Mining

The protocol vulnerability mining model based on reinforcement learning is used to improve the accuracy of mining multi-source power IoT terminal code vulnerabilities, as well as the coverage and efficiency, so as to ensure the security of the power grid, as shown in Figure 3.

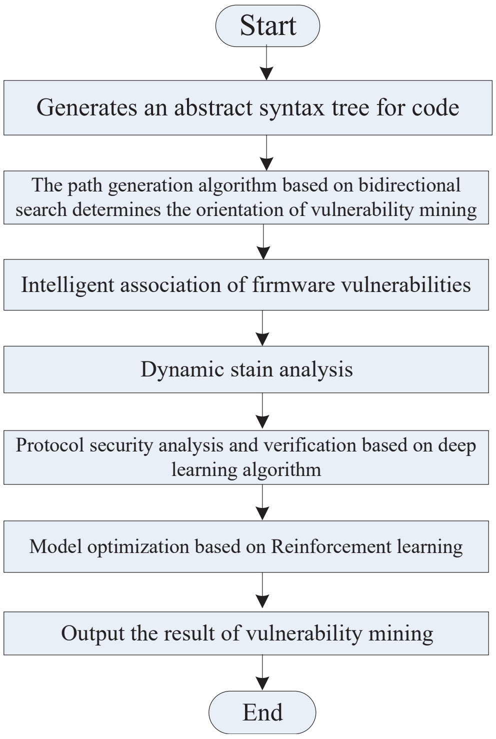

Start
Generates an abstract syntax tree for code
The path generation algorithm based on bidirectional search determines the orientation of vulnerability mining
Intelligent association of firmware vulnerabilities
Dynamic stain analysis
Protocol security analysis and verification based on deep learning algorithm
Model optimization based on Reinforcement learning
Output the result of vulnerability mining
End

Figure 3: Protocol vulnerability mining model based on reinforcement learning

Using the common lexical analysis and syntax analysis methods in the static analysis, the source code of the multi-source power IoT terminal is scanned and analyzed, and the abstract syntax tree of the code is constructed. The abstract syntax tree is used as a sample set. The path generation algorithm based on two-way search automatically obtains all program paths from the binary code input position sequence to its code area, and expands the area where the source code vulnerabilities are mined. It also studies the intelligent association of firmware vulnerabilities of multi-source power IoT terminals, and uses dynamic stain analysis technology to locate the possible location of firmware vulnerabilities, providing basic support for the subsequent code vulnerability mining. The protocol security is analyzed and verified based on the deep learning algorithm, and the mining module is optimized through reinforcement learning to complete the mining of code vulnerability.

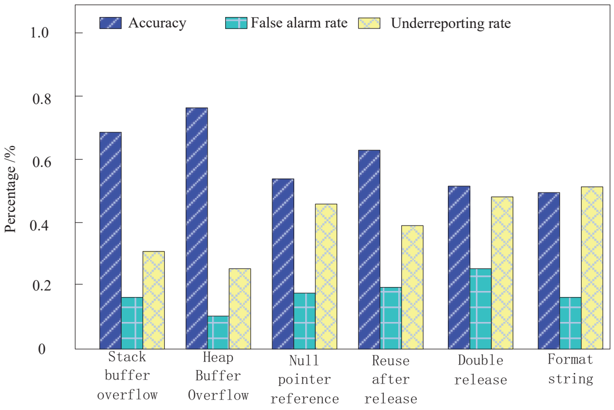


Figure 4: The accuracy rate, false alarm rate and underreporting rate of this method

3 Experiment and Analysis

3.1 Experimental Environment

The Juliet test suite is used as the test set to verify the vulnerability mining effect of proposed method, which contains test cases for C/C++ and Java. Each test case presents a code vulnerability defined in CWE (common vulnerability enumeration) in a simple manner. The entire test set contains about 57000 C/C++ test cases and 24000 Java test cases, covering 118 different types of vulnerabilities. Each sample code contains functions with and without vulnerabilities. We choose the test cases for C/C++ to test the accuracy of our tools.

In the experiment, four types of vulnerabilities including buffer overflow vulnerability, reuse after release, null pointer reference and format string are selected for the source code of the multi-source power IoT terminal. Among them, we subdivide the buffer overflow vulnerability into a stack based and heap based, and test the double release as a separate category. The quantity distribution is shown in Table 1. To characterize the efficiency of this method for vulnerability mining, the three commonly used indicators of correctness, false alarm rate and underreporting rate are used. The final test results are shown in Table 2.

3.2 Quantitative Results

According to the defined indexes, we calculate the correctness rate, false alarm rate and underreporting rate of the proposed method. The overall comparison is shown in Figure 4.

It can be seen from Figure 4 that the detection accuracy rate of stack buffer overflow, heap buffer overflow and reuse after release in the proposed method is high, all higher than 60%. Although the detection rate of the other four vulnerabilities is relatively low, they are all above 50%, and will not be completely undetectable.

False alarm rate and underreporting rate are two very

Table 1: Number of test cases for each type of vulnerability

Vulnerability type	Test case name	Quantity
Stack buffer overflow	CWE141_Stack_Based_Buffer_Overflow/s08	629
Heap buffer overflow	CWE172_Heap_Based_Buffer_Overflow/s03	113
Null pointer reference	CWE376_NULL_Pointer_Dereference	373
Reuse after release	CWE356_Use_After_Free	151
Double release	CWE457_Double_Free/s01	337
Format string	CWE144_Uncontrolled_Format_String/s06	617

Table 2: Final test results

Vulnerability type	Number of test cases	Number of tests	Correct quantity
Stack buffer overflow	629	535	438
Heap buffer overflow	113	99	87
Null pointer reference	373	272	209
Reuse after release	151	121	91
Double release	337	259	179
Format string	617	413	313

important indicators in vulnerability detection. If the false alarm rate is too high, it will bring a lot of unnecessary workloads. If the underreporting rate is too high, it will not achieve the effect of vulnerability detection. Therefore, good vulnerability mining work will have a balance between these two indicators. It can be seen from Figure 4 that the false alarm rate of the proposed method is relatively low, with an average of about 20%, also it has a high false alarm rate for some vulnerabilities, the highest is 49.2%, and the lowest is 23%. However, the proposed method performs well in the accuracy and false alarm rate. It can detect most vulnerabilities without a high false alarm rate, but it also has the problem of a high false alarm rate for some vulnerabilities.

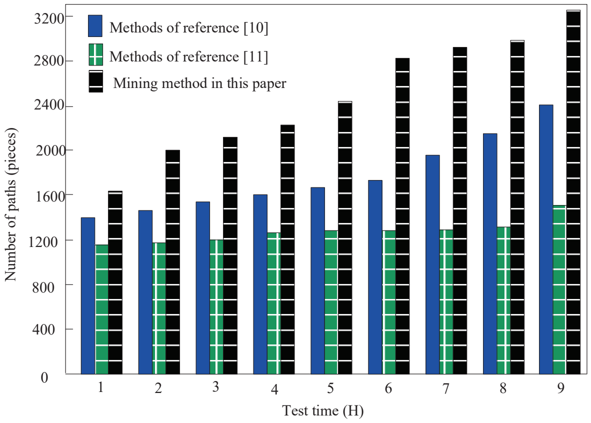


Figure 5: Perform the new path number test results

3.3 Number of New Paths Executed by Vulnerability Mining

The experiment analyzes the coverage rate of the code and the vulnerability mining situation. The vulnerability mining method based on the anti-sample algorithm [11] and the botnet vulnerability mining method based on machine learning [5] are compared with the proposed methods. The details are shown in Figure 5.

It can be seen from the Figure 5 that during the 8-hour test, the number of new paths executed by the proposed method is 3212, while the number of [11] is 2395, and the number of [5] is only 1532, and no more paths are added in the last 3 hours. The proposed method performs more new paths than the comparison methods, and the growth rate of the new paths is also higher in unit time, which indicates that the proposed method can mine more vulnerabilities on the paths in a shorter time, enhance the detection ability of its code branches, and have a higher test ability.

3.4 Code Coverage Test

The code coverage of proposed method and the two comparison methods is shown in Figure 6, where the code coverage is defined as the ratio of the number of unexecuted code basic blocks to the total number of code basic blocks. According to Figure 6, in the 8-hour test, the code coverage rate of proposed method is 28.7%, the code coverage rate of the method in [11] is 32.3%, and the final code coverage rate of the method in [5] is 13.6%. In addition, the code coverage rate of proposed method is higher than that of the comparison methods, which indicates that proposed method can achieve the optimal code

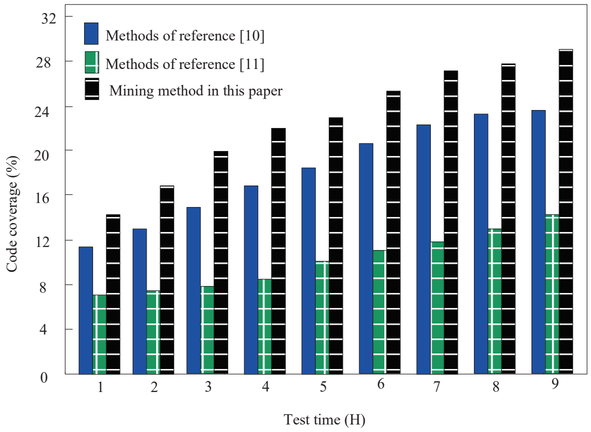


Figure 6: Code coverage test results

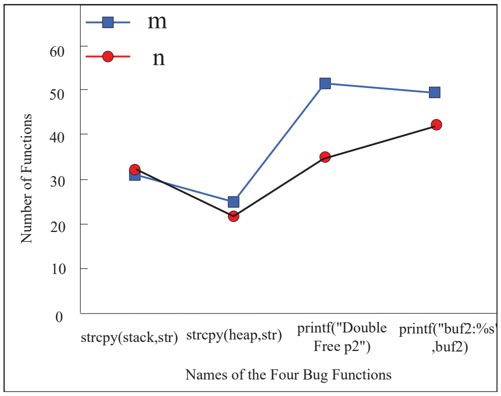
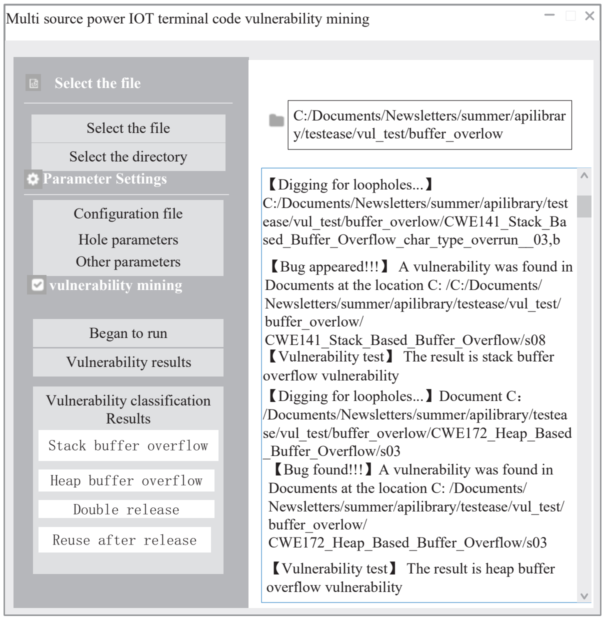


Figure 7: Searching results of four vulnerability functions in routers



C:/Documents/Newsletters/summer/apilibrary/testease/vul_test/buffer_overlow

【Digging for loopholes...】 C:/Documents/Newsletters/summer/apilibrary/testease/vul_test/buffer_overlow/CWE141_Stack_Based_Buffer_Overflow_char_type_overrun__03,b

【Bug appeared!!!】 A vulnerability was found in Documents at the location C: /C:/Documents/Newsletters/summer/apilibrary/testease/vul_test/buffer_overlow/CWE141_Stack_Based_Buffer_Overflow/s08
【Vulnerability test】 The result is stack buffer overflow vulnerability
【Digging for loopholes...】Document C：/Documents/Newsletters/summer/apilibrary/testease/vul_test/buffer_overlow/CWE172_Heap_Based_Buffer_Overflow/s03
【Bug found!!!】A vulnerability was found in Documents at the location C: /Documents/Newsletters/summer/apilibrary/testease/vul_test/buffer_overlow/CWE172_Heap_Based_Buffer_Overflow/s03

【Vulnerability test】 The result is heap buffer overflow vulnerability

Figure 8: Graphical interface for code vulnerability mining of multi-source power IoT terminal

coverage rate in a shorter time, and is conducive to the subsequent multi-source power Internet of things terminal code vulnerability mining.

3.5 Search Test of Four Vulnerability Functions

The experiment mainly verifies whether the method has a good correlation effect in the real situation, that is, using the real vulnerability function to correlate in the real firmware. The experimental process is as follows:

1) Select four vulnerability functions commonly used in the firmware of multi-source power IoT terminal, as shown in Table 3.

2) 375 firmware of MIPs platform are randomly selected from the firmware of multi-source power IoT terminal. The screening condition is whether the file name contains elf files of "CGI" and "Web" substrings, and a total of 305 binary files and 75113 functions are obtained. Among them, the number of functions with the same name as the vulnerability function is 30, 25, 50 and 48 respectively.

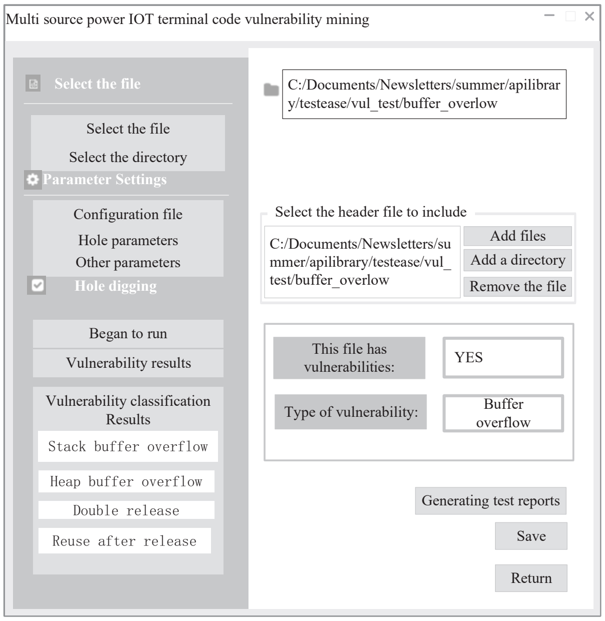
3) The numerical similarity between each vulnerability function and the 74006 functions in step (1) is calculated and sorted. Assuming n is the number of functions with the same name as the vulnerability function among the 75113 functions, and m is the number of functions with the same name among the top n functions with the highest score. Ideally, the numerical similarity of the top n functions with the same name with the highest score ranks among the 75113 functions. The value of M is between 0 and N, and the closer m is to N, the better the correlation effect. Experiments show that the number of the first n functions with the highest scores is 31, 21, 35, 42. The correlation results of the four vulnerability functions in the multi-source power IoT terminal firmware are shown in Figure 7, which shows that the proposed method has a good vulnerability association effect and provides a basis for subsequent vulnerability mining.

The experiment uses Python language as the code vulnerability mining tool for power terminals. The tool can be started in a graphical interface or command line mode, as shown in Figure 8. The function area is on the left side of the interface which displays various functions. The top of the interface is the file display box, which is used to display the testing file selected by the user. At the bottom of the interface is the vulnerability mining result display box, which shows the output information and the final result during the code vulnerability mining process.

The vulnerability mining interface is mainly divided into two parts: the left is the toolbar and the right is the specific functional area. The toolbar contains three tabs,

Table 3: Four vulnerability functions in multi source power IOT terminal firmware

Vulnerability type	Bug Function	Firmware	Binary
Stack buffer overflow	strcpy(stack,str)	B860AV2.1-A	str
Heap Buffer Overflow	strcpy(heap,str)	ZTE-B860AV1.1T	str
Null pointer reference	printf("Double Free p2")	WDR3320v2	free
Reuse after release	printf("buf2",buf2)	DIR-815V101	buf2



C:/Documents/Newsletters/summer/apilibrary/testease/vul_test/buffer_overlow

Select the header file to include

C:/Documents/Newsletters/summer/apilibrary/testease/vul_test/buffer_overlow

This file has vulnerabilities: YES

Type of vulnerability: Buffer overflow

Figure 9: Parameter setting of power terminal code vulnerability mining

namely "select file", "parameter setting" and "vulnerability mining". "Select file" tab is mainly used to select and clear the files to be tested. "Parameter setting" tab is mainly used for parameter setting, such as the "file configuration" button can set the header file and user-defined configuration file to be included in the code file to be detected. Click the "start running" button under the "vulnerability mining" tab, all files and parameters will be set by default, and the vulnerability mining process will be started. Click the buttons of the "parameter setting" tab in the left toolbar, the function area will display the corresponding parameter selection interface, as shown in Figure 9.

This tool scans and analyzes the input C language program, generates the abstract syntax tree of the code, and checks the abstract syntax tree model. The basis for model checking comes from the code vulnerability pattern described in Chapter 3. Once the code is detected to match a certain vulnerability pattern, it is judged that the vulnerability may exist, the vulnerability type and location are recorded, and a vulnerability report is generated at the end.

4 Discussion

The code vulnerability mining method of multi-source power IoT terminal based on reinforcement learning can detect the code vulnerability of the power terminal, but there are still shortcomings and room for improvement, which are mainly reflected in the following aspects:

1) The proposed method only implements the detection of the four most common vulnerabilities, but in actual application, there are other types of code vulnerabilities. We will continue to analyze and support the detection of more types of vulnerabilities.

2) Due to the lack of test samples, the testing of the software code of the multi-source power IoT terminal in this paper is not comprehensive enough. We will collect more multi-source power terminal software code samples for more testing in the future.

3) The proposed method also faces the problems of code complexity index rising and state space explosion when detecting large-scale programs. The problem is the biggest defect of the current program static analysis method. How to optimize and improve it needs further research.

5 Conclusion

In the smart grid environment, the access of a large number of heterogeneous power terminals will bring security risks and hidden dangers. Most of the power terminal devices are embedded devices, which have software vulnerabilities, and because the embedded devices have defects such as difficult software maintenance and long update cycle, the vulnerabilities in these power terminals will be latent for a long time and difficult to be repaired in time. This paper proposes a multi-source power IoT terminal code vulnerability mining method based on reinforcement learning to improve the speed and accuracy of vulnerability mining. It also expands the coverage of vulnerability mining and improves the security of the smart grid.

Acknowledgments

This work was supported by National Natural Science Foundation of China under grant NO. 62162026.

References

[1] A. Agazzi, J. Lu, "Global optimality of softmax policy gradient with single hidden layer neural networks in the mean-field regime" *ICLR*, vol. 35, no. 21, pp. 187-195,2021.

[2] T. Alladi,V. Chamola,B. Sikdar, and K. R. Choo,"Consumer IoT: Security vulnerability case studies and solutions," *IEEE Consumer Electronics Magazine*, vol. 9, no. 2, pp. 17–25, 2020.

[3] F. B. Allyson, M. L. Danilo, S. M. Jose and B. C. Giovanni,"Sherlock n-overlap: invasive normalization and overlap coefficient for the similarity analysis between source code," *IEEE Transactions on Computers*, vol. 68, no. 5, pp. 740–751, 2018.

[4] H. S. Chen, M. Pendleton, L. Njilla and S. H. Xu,"A survey on ethereum systems security: vulnerabilities, attacks, and defenses," *ACM Computing Surveys*, vol. 53, no. 3, pp. 1–43, 2020.

[5] Z. Chu, H. Yi, and K. Zhao,"Botnet vulnerability intelligence clustering classification mining and countermeasure algorithm based on machine learning," *IEEE Access*, vol. 7, pp. 182309–182319, 2019.

[6] J. Duan, D. Shi, R. S. Diao, H. F. Li, Z. W. Wang, B. Zhang, D. S. Bian, and Z. H. Yi,"Deep-reinforcement-learning-based autonomous voltage control for power grid operations," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 814–817, 2019.

[7] K. Filus, P. Boryszko, J. Domańska, M. Siavvas and E. Gelenbe,"Efficient feature selection for static analysis vulnerability prediction," *Sensors*, vol. 21, no. 4, pp. 1133, 2021.

[8] I. U. Haq and J. Caballero,"A survey of binary code similarity," *ACM Computing Surveys*,vol. 54, no. 3, pp. 1–38, 2021.

[9] M. S. Hwang, Y. L. Tang, C. C. Lee, "An efficient authentication protocol for GSM networks," *IEEE/AFCEA EUROCOMM 2000. Information Systems for Enhanced Public Safety and Security*, pp. 326-329, 2000.

[10] C. Konstantinou and S. P. Mohanty,"Cybersecurity for the smart grid," *Computer*, vol. 53, no. 5, pp. 10–12, 2020.

[11] Y. X. Lai, H. J. Gao, and J. Liu,"Vulnerability mining method for the modbus tcp using an anti-sample fuzzer," *Sensors*, vol. 20, no. 7, pp. 2040, 2020.

[12] C. C. Lee, M. S. Hwang, W. P. Yang, "Extension of authentication protocol for GSM," *IEE Proceedings – Communications*, vol. 150, no. 2, pp. 91-95, 2003.

[13] C. T. Li, M. S. Hwang, "A lightweight anonymous routing protocol without public key en/decryptions for wireless ad hoc networks",*Information Sciences*, vol. 181, no. 23, pp. 5333–5347, Dec. 2011.

[14] R. Q. Lin, Q. Luo,"Software Vulnerability Detection Algorithm Based on Deformable Convolutional Neural Network," *Computer Simulation*, vol. 38, no. 3, pp. 405–409, 2021.

[15] C. H. Ling, C. C. Lee, C. C. Yang, and M. S. Hwang, "A secure and efficient one-time password authentication scheme for WSN", *International Journal of Network Security*, vol. 19, no. 2, pp. 177-181, Mar. 2017.

[16] J. P. Men, G. Q. Xu, Z. Han, Z. H. Sun, X. J. Zhou, W. J. Lian and X. C. Cheng,"Finding sands in the eyes: vulnerabilities discovery in IoT with EUFuzzer on human machine interface," *IEEE Access*, vol. 7, pp. 103751–103759, 2019.

[17] V. R. Shen,"Novel code plagiarism detection based on abstract syntax tree and fuzzy Petri nets," *International Journal of Engineering Education*, vol. 1, no. 1, pp. 46–56, 2019.

[18] S. Siboni, V. Sachidananda, Y. Meidan, M. Bohadana, Y. Mathov, S. Bhairav, A. Shabtai and Y. Elovici,"Security testbed for Internet-of-Things devices," *IEEE transactions on reliability*, vol. 68, no. 1, pp. 23–44, 2019.

[19] H. Sun, C. Zhang, H. Li, Z. H. Wu, L. F. Wu and Y. Li,"ATOS: Adaptive program tracing with online control flow graph support," *IEEE Access*, vol. 7, pp. 127495–127510, 2019.

[20] J. F. Wang and P. Srikantha,"Stealthy black-box attacks on deep learning non-intrusive load monitoring models," *IEEE Transactions on Smart Grid*, vol. 12, no. 4, pp. 3479–3492, 2021.

[21] K. H. Wu, J. W. Li and B. Zhang,"Abnormal detection of wireless power terminals in untrusted environment based on double hidden Markov model," *IEEE Access* , vol. 9, pp. 18682–18691, 2020.

[22] S. Wu, W. Hu, L. Zhang and X. Y. Liu,"An intelligent key feature selection method of power grid based on artificial intelligence technology," *Zhongguo Dianji Gongcheng Xuebao/Proceedings of the Chinese Society of Electrical Engineering*, vol. 39, pp. 14–21, 2019.

[23] X. Xie, Y. Huang, W. Ning, *et al.*, "RDAD: A reconstructive and discriminative anomaly detection model based on transformer," *International Journal of Intelligent Systems*, vol. 37, no. 11, pp. 8928-8946,2022.

[24] X. Xie, X. Li, B. Wang, *et al.*, "Unsupervised abnormal detection using VAE with memory," *Soft Computing*, vol. 26, no. 02, pp. 1-13,2022.

[25] X. Xie, W. Ning, Y. Huang, *et al.*, "Graph-based Bayesian network conditional normalizing flows for multiple time series anomaly detection," *International Journal of Intelligent Systems*, vol. 15, no. 12, pp. 84–92, 2022.

[26] Y. X. Xue, Z. Z. Xu, M. Chandramohan and Y. Liu,"Erratum to "Accurate and Scalable Cross-Architecture Cross-OS Binary Code Search with Emulation"," *IEEE Transactions on Software Engineering*,vol. 47, no. 5, pp. 1088, 2021.

[27] M. Yi , X. H. Xu and L. Xu,"An intelligent communication warning vulnerability detection algorithm based on iot technology," *IEEE Access*, vol. 7, pp. 164803–164814, 2019.

[28] Y. H. Zhang, F. Y. Ren, A. X. Wu, T. T. Zhang, J. Cao and D. Zheng,"Certificateless multi-party au-

thenticated encryption for NB-IoT terminals in 5G networks," *IEEE Access*, vol. 7, pp. 114721–114730, 2019.

Biography

Hao Yang. He received his master's degree in computer software and theory from Huazhong University of Science and Technology in 2005. He's currently employed at State Grid Jiangxi Electric Power Research Institute. His research interests include software theory, network and information security.

Jun Li. He accomplished master degree in digital signal processing, University of York in 2012. He is employed by State Grid Jiangxi Electric Power Co., Ltd. His research interests include Software engineering,network and information security.

Junfeng Zhang. He received his master's degree in power systems and automation from Huazhong University of Science and Technology in 2008. He is employed by State Grid Jiangxi Electric Power Co. Ltd. His research interests include network and information security,power system automation, new digital technology.

Xin Xie. He received his master's degree in Control Theory and Control Engineering of Nanchang University in 2001. He is employed by East China Jiaotong University. His research interests include computer vision, computer network and information security.

Research on Secure Storage of Electronic Data through Blockchain Technology

Geng Niu
(Corresponding author: Geng Niu)

Shaanxi Police College, Xi'an, Shaanxi 710021, China
Email: gn67iq@126.com

(Received Apr. 2, 2022; Revised and Accepted Mar. 23, 2023; First Online Apr. 27, 2023)

Abstract

This article briefly introduced blockchain technology and the electronic data secure storage model based on blockchain. First, it introduced an incentive mechanism in the Practical Byzantine Fault Tolerance (PBFT) consensus algorithm used in the model to optimize it. Then, simulation experiments were conducted on the electronic data secure storage model. Finally, it was compared with the electronic data secure storage models under the Proof of Work (PoW) and Proof of Stake (PoS) consensus algorithms. The results showed that the electronic data secure storage model proposed in this article commonly used electronic data on-chain; the model effectively detected data tampering behavior on the local server and restored the tampered data. Furthermore, the improved PBFT algorithm maintained a higher data throughput and lower average latency at a higher request transmission rate.

Keywords: Blockchain; Consensus Algorithm; Electronic Data; Secure Storage

1 Introduction

With the development of Internet technology and the popularization of mobile devices, people's online lives are becoming more and more enriched, and activities that were originally conducted offline, such as daily work communication, shopping, and renting, can now be done online [17] Although the convenience has greatly improved, it has also brought the disputes that would have arisen offline to the online world. When the parties involved or a third party mediate or arbitrate disputes, evidence is needed just like in offline activities [1]. The evidence for online activities is electronic data generated during the Internet interaction process. However, compared with traditional offline physical evidence, electronic evidence is more easily tampered with or destroyed, and if electronic data is to be treated as legitimate evidence, it is necessary to ensure the security of electronic data storage [11–13, 20].

Blockchain technology has the features of decentralization, tamper-proofing, and easy-to-trace, which are in line with the secure storage need of electronic data. Related studies are shown below [3, 4, 7, 23]. Tian [25] studied the application of radio frequency identification and blockchain technology in constructing agricultural product supply chains. They collected, transmitted, and shared real data on product production, treatment, storage, distribution, and sales to ensure the authenticity of food information [5, 14, 21].

Yi [26] proposed the application of blockchain technology to protect logistics security and personal privacy, constructed a logistics blockchain model, and verified its efficiency and security on a distributed platform. Mao *et al.* [22] used blockchain technology to enhance the effectiveness of supervision and management in the food supply chain and evaluated the credit evaluation text collected by the blockchain using Long Short-Term Memory (LSTM). The results suggested that LSTM with blockchain performed better for credit evaluation text. This article briefly introduced blockchain technology and the electronic data secure storage model based on blockchain. An incentive mechanism was introduced to the Practical Byzantine Fault Tolerance (PBFT) consensus algorithm to improve the model. The electronic data secure storage model was then simulated and compared with the electronic data secure storage models under the Proof of Work (PoW) and Proof of Stake (PoS) consensus algorithms.

2 Blockchain

Figure 1 shows the basic architecture of a block in a blockchain [2,6,19]. A block consists of a head and a body. The head is an essential component of the block, which seals the version number, random number, timestamp, previous block hash, current block hash, and Merkle root of the block [24]. The block body contains data related to the transactions involved in the block, and they are ensured to be authentic and tamper-proof through digital signatures and summarized into a Merkle root using a hash function. In the use of blockchain, the block body data between blocks does not directly interact with each other, but rather interact through the hash function in the head, to verify whether the data stored in the block has been tampered with [9].

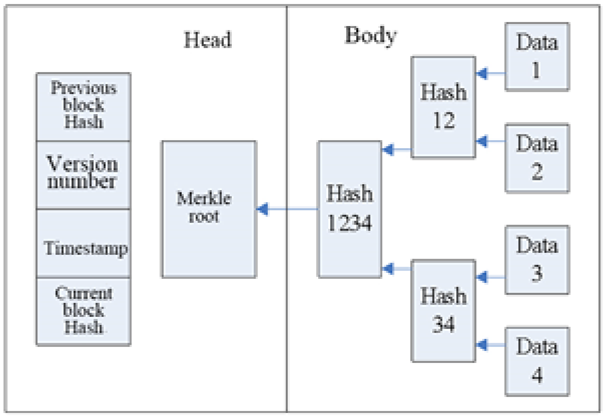


Figure 1: The basic architecture of the block in blockchain

3 Blockchain-based Electronic Data Secure Storage

3.1 Basic Model of Electronic Data Storage

The basic architecture of the model used for secure storage of electronic data combined with blockchain is shown in Figure 2. In terms of hierarchy, it is separated into application service layer, data storage layer, smart contract layer, consensus mechanism layer, and blockchain data layer. The application service layer is the structural layer that directly interacts with users [10], which mainly includes functions such as user registration and login, data submission, data finding, and security management. The data storage layer is usually a local central database or a trusted third-party database for storing user data, uploaded complete data, log data for operations on the storage system, etc.. The smart contract layer contains smart contracts with various execution functions, including encryption of stored data and permission management of the storage system. The consensus mechanism layer contains the consensus mechanism algorithm for synchronizing the data of blockchain nodes, which is the focus of the entire blockchain electronic data storage. The blockchain data layer contains plural nodes for building blocks [18].

The entire blockchain electronic data storage model includes complete electronic evidence information, user data, platform business data, and operation log data. The complete electronic evidence information is stored in a central database or a trusted third-party database, while user data and platform business data are stored in the platform's own database. The hash data of the electronic evidence and operation log data are uploaded to the blockchain for storage after being verified by the consensus algorithm. It can be said that the storage of electronic evidence, operation log data, and user and platform business data are independent of each other. In a separate platform, only platform business data and user traceability can be queried, and it is impossible to directly obtain complete electronic evidence. After the operation log data is uploaded to the chain, it receives the endorsement of blockchain technology, ensuring the accuracy and immutability of the log data [8].

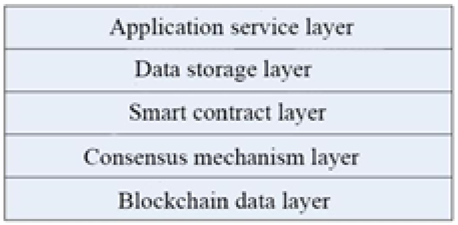


Figure 2: Basic architecture of the blockchain-based secure storage model for electronic data

3.2 Process of Blockchain Electronic Data Storage

As described in the previous section on the model for blockchain electronic data storage, the platform does not contain complete electronic evidence [16], and queries can only be made after account permissions have been verified. During this process, the log data generated by the operations will be uploaded to the blockchain for storage along with the hash value of the electronic data.

Figure 3 shows the secure storage process for electronic data based on blockchain. The steps are shown below.

1) The user logs in to the secure storage system for electronic data with their account password.

2) In the storage system model, the user navigates to the storage page, submits the complete electronic data at the corresponding input point, selects a storage location for the data, and stores the complete data. The storage location includes the local center database and trusted third-party databases.

3) The user selects the default wallet or provides their personal wallet, which is used for encrypting the data and is related to the user's permissions.

4) The user encrypts the electronic data using the key provided by the wallet and generates the digest information of the complete electronic data, i.e., the hash value, along with the corresponding operation log data.

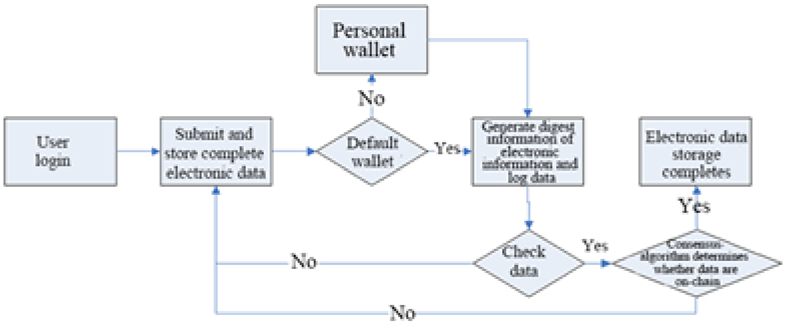

Personal wallet
User login
Submit and store complete electronic data
No
Default wallet
Yes
Generate digest information of electronic information and log data
Electronic data storage completes
Yes
Check data
Yes
Consensus algorithm determines whether data are on-chain
No
No

Figure 3: The secure storage process of electronic data based on blockchain

5) The digest information and log data are verified using the pre-set rules of the smart contract. If the verification fails, the user needs to re-enter the electronic data. If the verification passes, the process proceeds to the next step.

6) Whether the data can be added to the blockchain is determined by a consensus algorithm. In simple terms, the consensus algorithm broadcasts the data to be added to the blockchain from one node to other nodes in the blockchain for verification. When the majority of nodes have verified the data, it will be added to the blockchain and backed up in other nodes. This article uses the PBFT algorithm as the consensus algorithm to determine whether the data can be added to the blockchain. In the PBFT algorithm, it is first necessary to determine a primary node in the same view of the blockchain, with other nodes in the view serving as secondary nodes. The formula for selecting the primary node is:

$$p = v \mod |n|, \tag{1}$$

where p is the serial number of the primary node, v is the serial number of view, and $|n|$ is the total number of nodes in the view.

The traditional PBFT algorithm assumes that all normal nodes will participate in the cooperation, but in reality, nodes in a blockchain network tend to be rational and self-interested. In order to maximize their own interests, they are more inclined not to cooperate, resulting in a low success rate of consensus among blockchain network nodes, which is not conducive to the storage of electronic data. Therefore, this paper introduces an incentive mechanism when selecting the primary node [27]. After selecting primary node p through Equation (1), the incentive value of the node must also be judged. Only when the incentive value of the node exceeds the preset threshold can it be selected as the primary node. The formula for calculating the incentive value that the node can obtain after each successful consensus is reached is:

$$\begin{aligned} \theta_1 &= \alpha \cdot \theta \\ \theta_{GC} &= \begin{cases} \frac{\theta}{n} - c & \text{node cooperation} \\ \frac{\theta_1}{n} & \text{node non-cooperation} \end{cases} \end{aligned} \tag{2}$$

where $theta$ is the service fee paid by the client to the blockchain network, α is the proportional factor of the service fee for revenue lure to the node, $theta_1$ is the service fee for revenue lure to the node, n is the number of nodes in the blockchain, c is the cost per node to participate in the collaboration, and $theta_{GC}$ is the incentive value (net gain) that each node can obtain after successful consensus, which varies depending on whether this consensus process is collaborative or not. After determining the primary node using Equations (1) and (2), the client sends a data-on-chain request to the primary node. After the primary node verifies the request, it broadcasts it to the nodes in the blockchain. The secondary nodes validate the request and provide feedback on the validation result to the other nodes in the blockchain. When more than two-thirds of the nodes pass the validation, the consensus is successful, and each node stores the uploaded data in its local blockchain ledger, completing the secure storage of electronic data; otherwise, the client is notified to resend the request.

4 Experiment Analysis

4.1 Test Environment

The testing of the blockchain-based electronic data secure-storage model was conducted on the laboratory server, and the blockchain network required for the electronic data storage model was provided by the Ethereum virtual machine. The laboratory server used for testing had the following specifications: quad-core i7 CPU, 16 GB memory, and 1024 GB hard disk. To facilitate the simulation, the Ethereum virtual machine was set with a single-core i5 CPU, a working frequency of 2.5 GHz, and 4 GB storage. The virtual machine provided ten nodes.

4.2 Testing Content

1) Test of electronic data on-chain function The storage on-chain function of the electronic data secure-storage model was tested first. The steps were:

   a. Login to the account;

b. Upload electronic data following the prompts on the electronic data upload interface;

c. Click the "on-chain" button;

d. Check the on-chain status of the data in the evidence list.

2) Test of electronic data storage security In the electronic data secure-storage model, the data was not only stored on the laboratory's server but also backed up in the nodes of the blockchain network. Since the process of storing electronic data not only used account password permissions but also encrypted the data, it would take a long time for an attacker to modify the data on the laboratory's server. Therefore, in this security testing, the stored data on the laboratory's server was directly modified to simulate the scenario of the laboratory's server being attacked and tampered with. After that, the stored data was queried in the query interface of the electronic data secure-storage model.

3) Latency and throughput of electronic data storage model under different consensus algorithms For the electronic data secure-storage model based on blockchain, the consensus algorithm is an important mechanism to guarantee the security of electronic data on-chain. This paper introduced an incentive mechanism based on the traditional PBFT algorithm to encourage nodes to actively participate in cooperation. To further prove the performance of the improved PBFT algorithm, it was with the PoW and PoS algorithms in the experiment. The PoW algorithm allows nodes to compete for accounting rights by calculating difficult problems using computing power, and the nodes with accounting rights broadcast to the rest of the nodes. After verifying the broadcast computing results, the other nodes backup the block data of the accounting node to their local blockchain, thus keeping the data consistent in the distributed nodes of the blockchain.

The PoS algorithm introduces the concept of "coin age" based on PoW. It uses "coin age" to narrow the range of random values required for computing problems, reduce computing power, and make nodes with longer "coin age" easier to obtain accounting rights. The nodes that obtain accounting rights backup the block data, same as described above.

When testing the latency and throughput of the electronic data storage model under the three consensus algorithms, the size of the electronic data to be stored was set to 20 GB. Different data on-chain sending rates were then set to detect the average latency and data throughput of the electronic data storage model under the three consensus algorithms.

4.3 Experimental Results

This article first tested the data on-chain function of the blockchain-based electronic data security storage model. The steps for electronic data on-chain were divided into four steps, and the test results for each step are shown in Table 1.

Step 1 tested the login function of the storage model, and different feedback were given based on whether the account password was correct or not.

Step 2 tested the data upload function of the storage model, and whether the "upload" button could be used depended on whether there was input data in the input box on the upload page.

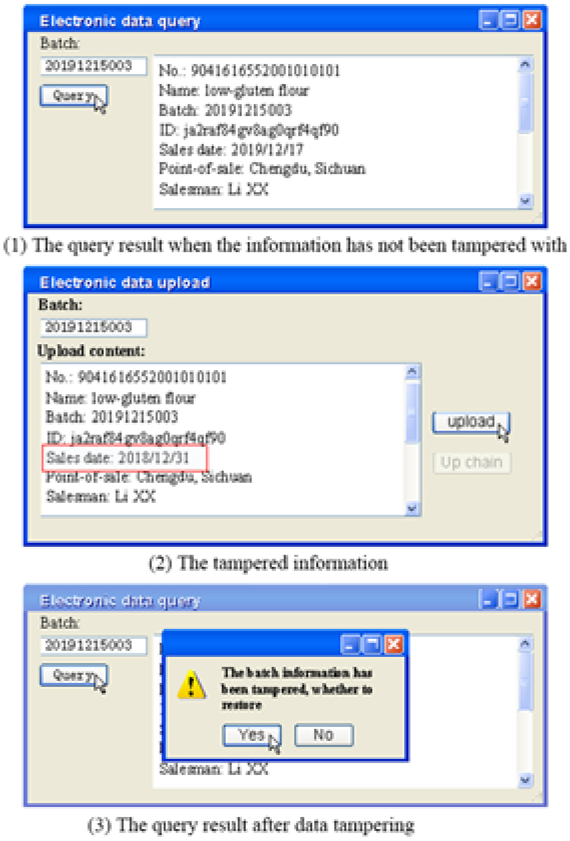
Step 3 tested the data on-chain function of the storage model, and whether the "on-chain" button could be used depended on whether the electronic data was successfully uploaded in the previous steps. Step 4 tested the query function of the storage model, and whether the data was successfully on-chain determined whether the corresponding electronic data could be queried in the query interface.

After testing the data on-chain function of the blockchain-based electronic data secure-storage model, security testing was conducted. The results are shown in Figure 4. In Figure 4, (1) is the data query result when the information has not been tampered with. (2) is the model upload interface when the storage data are tampered with by a hacker with permission, and the tampered content is highlighted by a red box. In this case, the hacker replaces "2019/12/17" in "sales date" with "2018/12/31". (3) is the query result after the data has been tampered with. It was seen from Figure 4 that when querying the tampered data, a prompt indicating that the data has been tampered with was returned due to the inconsistency between the data digest information stored in the local server and the backup data digest information on the blockchain, and it also asked if the data should be restored. The restored data content was consistent with (1), so it is not shown here in detail.

The consensus algorithm is an important component of the blockchain-based electronic data security storage model, which ensures the authenticity of data on-chain. In addition, the consensus algorithm requires nodes in the blockchain to reach consensus, which affects the efficiency of data on-chain. Figure 5 shows the data throughput and average latency of the storage model under different on-chain request sending rates for three consensus algorithms. First, regarding the change in data throughput, it was seen from Figure 5 that the throughput of three consensus algorithms increased with the increase of the sending rate when the request sending rate was low. However, after reaching a certain sending rate, the throughput remained basically unchanged. The PoW algorithm stabilized at a throughput of 230 TPS after reaching a sending rate of 500 TPS; the PoS algorithm stabilized at

Table 1: The testing results of the electronic data on-chain function

Testing process	Test results
Step 1	1: After entering the correct account password, the user logs in successfully and jumps to the data upload page. 2: If the account password is wrongly entered, the page will not jump, but the prompt of ”account or password error” will appear.
Step 2	1: Enter electronic data in the input box on the data upload page, click the ”upload” button, and return the prompt f ”successful upload”. 2: If electronic data are not entered in the input box, the ”upload” button cannot be clicked.
Step 3	1: After the electronic data is uploaded successfully, click the ”on-chain” button, and the prompt of ”successful on-chain” will appear. 2: When the electronic data upload fails, the ”on-chain” button cannot be clicked.
Step 4	1: In the query interface, the successfully on-chain data can be seen according to the serial number of electronic data. 2: In the query interface, the electronic data that are not successfully uploaded cannot be seen.



Batch: 20191215003

No.: 90416165520010101010
Name: low-gluten flour
Batch: 20191215003
ID: ja2raf84gv8ag0qrf4qf90
Sales date: 2019/12/17
Point-of-sale: Chengdu, Sichuan
Saleman: Li XX

(1) The query result when the information has not been tampered with

Batch: 20191215003

Upload content:

No.: 90416165520010101010
Name: low-gluten flour
Batch: 20191215003
ID: ja2raf84gv8ag0qrf4qf90
Sales date: 2018/12/31
Point-of-sale: Chengdu, Sichuan
Saleman: Li XX

(2) The tampered information

Batch: 20191215003

The batch information has been tampered, whether to restore

Saleman: Li XX

(3) The query result after data tampering

Figure 4: Electronic data storage security test results

a throughput of 320 TPS after reaching a sending rate of 600 TPS; the improved PBFT algorithm stabilized at a throughput of 450 TPS after reaching a sending rate of 800 TPS. Then, regarding the change in average latency, similarly, when the request sending rate was low, the average latency of the three consensus algorithms remained basically unchanged. However, after reaching a certain sending rate, the average latency increased. The PoW algorithm stabilized at an average latency of 0.56 s before reaching a sending rate of 500 TPS and then gradually increased; at a sending rate of 1000 TPS, the average latency was 3.24 s. The PoS algorithm stabilized at an average latency of 0.34 s before reaching a sending rate of 600 TPS and then gradually increased; at a sending rate of 1000 TPS, the average latency was 2.13 s. The improved PBFT algorithm stabilized at an average latency of 0.12 s before reaching a sending rate of 800 TPS and then gradually increased; at a sending rate of 1000 TPS, the average latency was 1.12 s.

5 Conclusion

This article briefly introduced blockchain and the electronic data secure-storage model based on blockchain. An incentive mechanism was introduced to improve the PBFT consensus algorithm used in the model. Then, simulation experiments were conducted on the electronic data secure-storage model. The improved model was compared with the electronic data secure-storage models under the PoW and PoS consensus algorithms. The results are as follows.

1) The test results of the electronic data secure-storage model for the on-chain function of electronic data showed that the model could perform normal on-

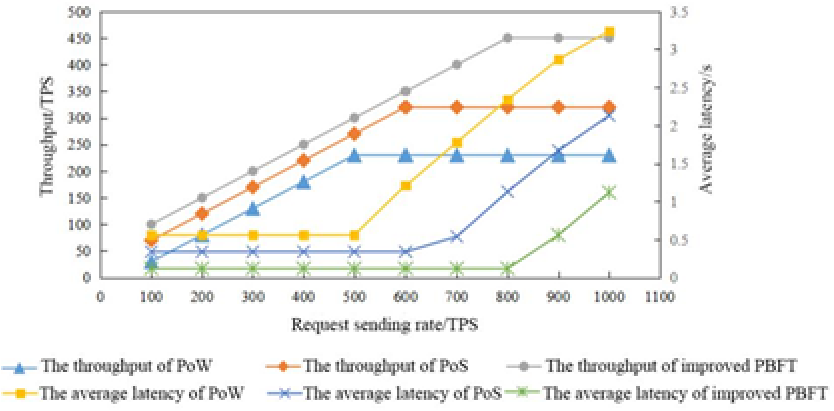


Figure 5: Average latency and data throughput of electronic data storage models under three consensus algorithms

chain operations on electronic data.

2) The electronic data secure-storage model could effectively detect whether the data has been tampered with and used the backup in the blockchain to restore the tampered data when facing local server data tampering.

3) Under the three consensus algorithms, the data throughput of the storage model increased with the increase of the request sending rate and remained stable after reaching a certain rate; the improved PBFT algorithm maintained higher throughput at higher sending rates; the average latency of data transmission remained stable before reaching a certain sending rate and then increased with the increase of the sending rate; the improved PBFT algorithm maintained a lower average latency at a higher sending rate.

References

[1] S. Ahmed, N. Ten Broek, "Blockchain could boost food security," *Nature*, vol. 550, no. 7674, pp. 43-43, 2017.

[2] A. Bhardwaj, G. Subrahmanyam, V. Avasthi, H. Sastry, "Ransomware: A Rising Threat of new age Digital Extortion," *Indian Journal of Science & Technology*, vol. 9, no. 14, pp. 1-6, 2015.

[3] P. Y. Chang, M.S. Hwang, C.C. Yang, "A blockchain-based traceable certification system", in *Security with Intelligent Computing and Big-data Services*, pp. 363-369, 2018.

[4] L. Chen, Q. Fu, Y. Mu, L. Zeng, F. Rezaeibagha, M. S. Hwang, "Blockchain-based random auditor committee for integrity verification", *Future Generation Computer Systems*, vol. 131, pp. 183-193, 2022.

[5] M. Y. Chen, C. W. Liu, M. S. Hwang, "Securedropbox: A file encryption system suitable for cloud storage services," in *Proceedings of the 2013 ACM Cloud and Autonomic Computing Conference*, pp. 1-2, 2013.

[6] Y. H. Chen, L. C. Huang, I. C. Lin, M. S. Hwang, "Research on the secure financial surveillance blockchain systems", *International Journal of Network Security*, vol. 22, no. 4, pp. 708-716, 2020.

[7] Y. H. Chen, L. C. Huang, I. C. Lin, M. S. Hwang, "Research on blockchain technologies in bidding systems", *International Journal of Network Security*, vol. 22, no. 6, pp. 897-904, 2020.

[8] P. Dangayach, "Pharmaceutical supply chain tracking system based on blockchain technology and radio frequency identification tags," *International Journal of Business Research*, vol. 19, no. 4, pp. 37-44, 2019.

[9] M. Dobrovnik, D. M. Herold, E. Fürst, S. Kummer, "Blockchain for and in Logistics: What to Adopt and Where to Start," *Logistics*, vol. 2, no. 3, pp. 1-14, 2018.

[10] M. Felder, M. Kuperberg, "Blockchain @ Deutsche Bahn: Anwendungen für mobility & logistics," *Dersenbahningenieur*, vol. 69, no. 2, pp. 14-16, 2018.

[11] W. F. Hsien, C. C. Yang and M. S. Hwang, "A survey of public auditing for secure data storage in cloud computing," *International Journal of Network Security*, vol. 18, no. 1, pp. 133-142, 2016.

[12] O. Hueber, "The blockchain and the sidechain innovations for the electronic commerce beyond the Bitcoin's framework," *International Journal of Transitions and Innovation Systems*, vol. 6, no. 1, pp. 88-102, 2018.

[13] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 5, 2017.

[14] M. S. Hwang, C. C. Lee, T. H. Sun, "Data error locations reported by public auditing in cloud storage service," *Automated Software Engineering*, vol. 21, no. 3, pp. 373–390, Sep. 2014.

[15] S. S. Kamble, A. Gunasekaran, M. Goswami, J. Manda, "A systematic perspective on the applica-

tions of big data analytics in healthcare management," *International Journal of Healthcare Management*, vol. 12, no. 3, pp. 226-240, 2019.

[16] L. Koh, A. Dolgui, J. Sarkis, "Blockchain in transport and logistics-paradigms and transitions," *International Journal of Production Research*, vol. 58, no. 7, pp. 2054-2062, 2020.

[17] N. Kshetri, "1 Blockchain's roles in meeting key supply chain management objectives," *International Journal of Information Management*, vol. 39, pp. 80-89, 2018.

[18] M. Li, S. Shao, Q. Ye, G. Xu, G. Q. Huang, "Blockchain-enabled logistics finance execution platform for capital-constrained E-commerce retail," *Robotics and Computer Integrated Manufacturing: An International Journal of Manufacturing and Product and Process Development*, vol. 65, pp. 1-14, 2020.

[19] C. Y. Lin, L. C. Huang, Y. H. Chen, M. S. Hwang, "Research on security and performance of blockchain with innovation architecture technology", *International Journal of Network Security*, vol. 23, no. 1, pp. 1-8, 2021.

[20] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for shared data storage with user revocation in cloud computing", *International Journal of Network Security*, vol. 18, no. 4, pp. 650–666, 2016.

[21] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *International Journal of Network Security*, vol. 18, no. 5, pp. 900–916, 2016.

[22] D. Mao, F. Wang, Z. Hao, H. Li, "Credit Evaluation System Based on Blockchain for Multiple Stakeholders in the Food Supply Chain," *International Journal of Environmental Research & Public Health*, vol. 15, no. 8, pp. 1-21, 2018.

[23] Z. P. Peng, M. L. Chiang, I. C. Lin, C. C. Yang, M. S. Hwang, "CBP2P: Cooperative electronic bank payment systems based on blockchain technology", *Journal of Internet Technology*, vol. 23, no. 4, pp. 683-692, 2022.

[24] S. Surjandy, Meyliana, A. N. Hidayanto, H. Prabowo, "The latest adoption blockchain technology in supply chain management: a systematic literature review," *ICIC Express Letters*, vol. 13, no. 10, pp. 913-920, 2019.

[25] F. Tian, "An agri-food supply chain traceability system for China based on RFID & blockchain technology," in *13th International Conference on Service Systems and Service Management (ICSSSM'16)*, pp. 1-6, 2016.

[26] H. Yi, "A secure logistics model based on blockchain," *Enterprise Information Systems*, no. 4, pp. 1-17, 2019.

[27] Y. Ziegler, V. Uli, F. Keller, A. Kramer, "The impact of blockchain networks on logistics: An update," in *Mednarodna konferenca o razvoju organizacijskih znanosti*, 2019.

Biography

Geng Niu, born in 1982, has received the master's degree from Xi'an University Of Science and Technology in 2007. She is working in Shaanxi Police College now as an associate professor. She is interested in cyber crime investigation and digital forensics.

Design and Analysis of Hash Function Based on Spark and Chaos System

Jiandong Liu, Yujie Liu, and Bo Li
(Corresponding author: Yujie Liu)

College of Information Engineering, Beijing Institute of Petrochemical Technology
Beijing 102617, P. R. China
Email: liuyujiechn@qq.com

(Received Oct. 6, 2022; Revised and Accepted Apr. 1, 2023; First Online Apr. 25, 2023)

Abstract

A hash function based on Spark and a two-dimensional coupled dynamic integer tent map is constructed to address the security and efficiency issues when handling large data volumes of plaintext information. First, the plaintext is read and partitioned in the Spark platform; then, each data block is processed in parallel; a Merkle tree structure is used to take fixed-length data every time for compression, and the compression function uses two-dimensional coupled image lattice with dynamic integer tent map, with additional dynamic parameters to enhance the obfuscation performance—the parallel processing by the Spark big data platform results in a significant improvement of the algorithm running efficiency.

Keywords: Chaos; Hash Function; Spark; Tent Map; Two-dimensional Coupled

1 Introduction

With the rapid development of Internet, cloud computing, Internet of Things, social media and other information technologies in recent years, the data accumulated in various industries are showing an explosive growth trend [12]. In order to cope with the situation such as the proliferation of data volume and to handle large-scale high-dimensional data, using Hadoop distributed computing is an effective solution, such as Kim *et al.* [10] and Cui *et al.* [4] proposed algorithms based on it. With the development of technology, people gradually found that MapReduce, the core component in Hadoop, has many drawbacks, such as not good at real-time computation and cannot return results in milliseconds or seconds like MySQL; the output results of each MapReduce job are written to disk, which will cause a lot of disk IO, resulting in very low performance. Thus Spark was born as an alternative to MapReduce, Spark implements a distributed fault-tolerant memory structure that uses as much memory as possible on multiple nodes, significantly improving overall performance, and many Spark-based algorithmic studies have emerged, leading to significant improvements in efficiency,such as Ji *et al.* [7]; Yin *et al.* [24]; Dries *et al.* [6]. However, with the increase in the ability to process data, the problem of securing sub aspects of data has gradually emerged, and there is an urgent need for a reasonable solution on how to secure large amounts of data.

Hash Function in cryptography are often used as security algorithms in network message transmission, and after classical Hash functions such as MD5 were broken, researchers found that applying chaos models to the design of Hash Functions can achieve very desirable results [2, 3, 9, 11, 13], after improving the security, it was found that because of the complexity of chaos systems, the efficiency of the algorithm could not reach the effect of traditional Hash functions, and also due to the development of computer hardware and the emergence of multi-core processors, Amir *et al.* [1] and Zhang *et al.* [27] focused their research on parallelized processing . But so far it is rare to see Hash functions designed based on chaotic systems implemented on Spark platform, while there are many related algorithms designed based on MapReduce instead. Zhai et al [25] implements the efficiency improvement of K-nearest neighbor algorithm based on hash technology and MapReduce big data platform, which substantially improves the efficiency of K-nearest neighbor algorithm while maintaining the classification capability for the problem of inefficiency or even infeasibility in big data environment; Zhang *et al.* [26] proposed a hash function based on column storage and MapReduce distributed connection algorithm for the traditional relational database in the operation of large data with serious degradation of system performance, introduced the MapReduce parallel computing model, and proved the good performance of its proposed algorithm in terms of execution time and load capacity through experiments. Including the literature presented above [4] [7] [24] [6] are all based on MapReduce or Spark proposed for the exploration of K-nearest neighbor algorithm.

The earliest chaos models that were applied and are still being explored so far are tent map and logistic map, and one-way hash function constructed by combining coupled image lattice map with tent map [17], This paper extends the one-dimensional coupled image lattice map to two- dimensional, maximizes the diffusion speed, takes the dynamic integer tent map as its lattice point calculation method, and then add dynamic parameters, and evolve the dynamic parameters by circular displacement method, which enhances the correlation between the explicit difference and the parameter term. The overall structure adopts merkle tree type, merkle tree is widely used in the distributed field, and its features such as fast hash recomputation and proof of existence of leaf nodes make it possible to quickly locate a small amount of changing data content under massive data, which plays a crucial role in blockchain, in turn, the hash algorithm is further adapted to distributed system computing, while the Spark big data platform is used for parallel implementation, which ensures security and improves efficiency at the same time.

2 Two-dimensional Dynamic Tent Map

2.1 Dynamic Integer Tent Map

Tent map is a classical nonlinear mapping with a general mechanism for generating chaos. Its expression is as follows.

$$F_\alpha : x_{n+1} = \begin{cases} \frac{x_n}{\alpha}, & 0 \le x_n < \alpha \\ \frac{1-x_n}{1-\alpha}, & \alpha \le x_n \le 1 \end{cases} \tag{1}$$

The sequence produced by this map exhibits chaotic behavior in the real number domain and has good uniform distribution properties. However, in the real number domain, with finite accuracy, cannot produce an acyclic orbit.

Simulation experiments are performed for Equation (1) in the double precision case. The results obtained are slightly different from the theoretical analysis due to the error introduced by the finite precision, but the final result is still 0 after only 50 or so iterations regardless of the initial value, and can be seen to be more prone to short-cycle orbits (as shown in Figure 1 and Figure 2).

The proof process is as follows:

When α=0.5, Equation (1) can be expressed as:

$$F_{0.5} : x_{n+1} = \begin{cases} 2x_n, 0 \le x_n < 0.5 \\ 2\left(1 - x_n\right), 0.5 \le x_n \le 1 \end{cases} \tag{2}$$

If $x_0 = (0.a_1a_2a_3 \ldots a_k000 \ldots)$, and $x_0 = 1$, then the result is 0 after at most $k+1$ iteration, that is,$x_m = 0 (m \ge k+1)$.

Prove: Use binary to represent the decimal between 0 and 1. The first iteration: According to the tent map definition, if $\bar{a}_k = 1 - a_k$,then,

$$f\left(x_0\right) = \begin{cases} \left(0.a_2a_3 \ldots a_k000 \ldots\right), a_1 = 0 \\ \left(0.\bar{a}_2\bar{a}_3 \ldots \bar{a}_k111 \ldots\right), a_1 = 1 \end{cases}$$

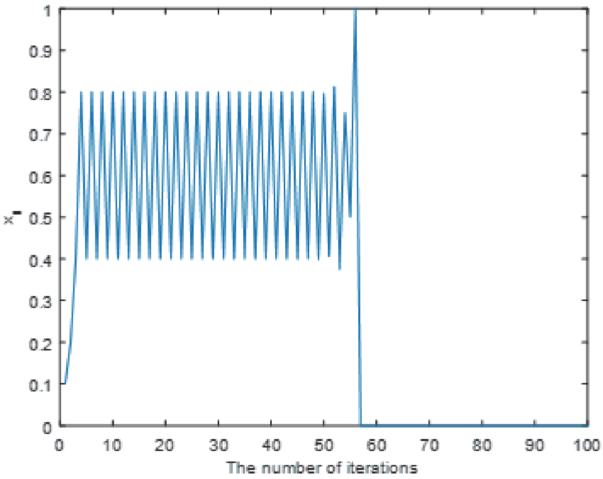


Figure 1: Iteration result $\alpha = 0.5\, x_0 = 0.1$

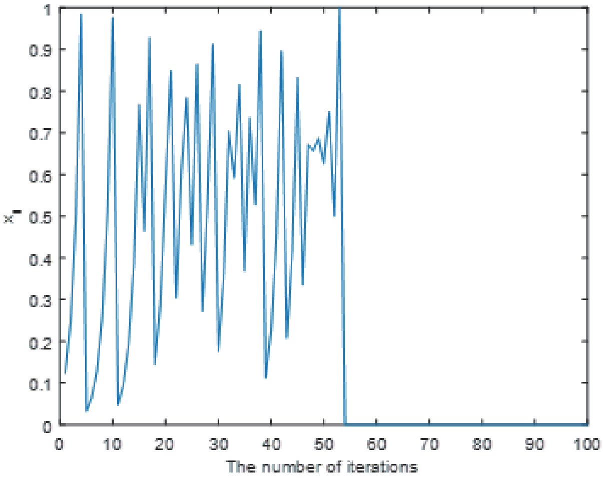


Figure 2: Iteration result $\alpha = 0.5\, x_0 = 0.123$

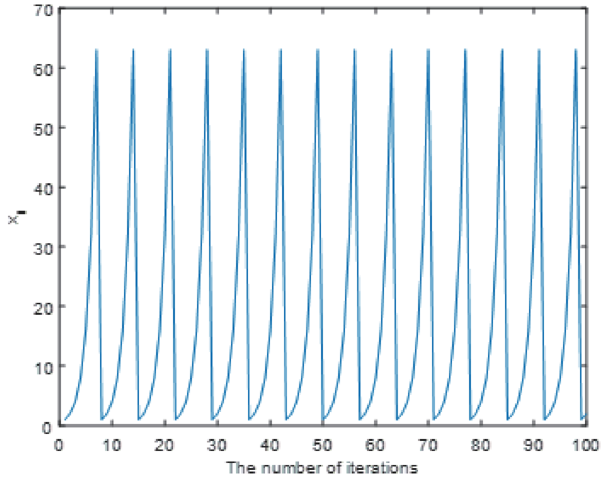


Figure 3: Integer tent map iteration

That is, when the first decimal of x_0 is 0(indicating that the number is less than 0.5), the calculated x_1 by tent map is equivalent to the decimal place of x_0 moved to the left one; When the first decimal of x_0 is 1(indicating that the number is greater than or equal to 0.5), then, after map, x_1 is equal to the decimal place of x_0 inverted, and then moved to the left one place.

The second iteration:
If $x_1 = (0.a_2a_3 \cdots a_k000\ldots)$,

$$f\left(x_1\right) = \begin{cases} \left(0.a_3 \ldots a_k000\ldots\right), a_2 = 0 \\ \left(0.\bar{a}_3 \ldots \bar{a}_k111\ldots\right), a_2 = 1 \end{cases}$$

If $x_1 = (0.\bar{a}_2\bar{a}_3 \ldots \bar{a}_k111\ldots)$,

$$f\left(x_1\right) = \begin{cases} \left(0.\bar{a}_3 \ldots \bar{a}_k111\ldots\right), \bar{a}_2 = 0 \\ \left(0.a_3 \ldots a_k000\ldots\right), \bar{a}_2 = 1 \end{cases}$$

The kth iteration:
If $x_{k-1} = (0.a_k000\ldots)$,

$$f\left(x_{k-1}\right) = \begin{cases} (0.000\ldots), a_k = 0 \\ (0.111\ldots), a_k = 1 \end{cases}$$

If $x_{k-1} = (0.\bar{a}_k111\ldots)$,

$$f\left(x_{k-1}\right) = \begin{cases} (0.111\ldots), \bar{a}_k = 0 \\ (0.000\ldots), \bar{a}_k = 1 \end{cases}$$

The $k+1$th iteration: $f\left(x_{k-1}\right) = (0.000\ldots)$.

Therefore, after at most $k+1$ times iteration, the result must be 0. This conclusion shows that the finite accuracy of the initial values determines the behavior of the iteration.

The tent map is equivalently transformed from a real domain operation to an integer domain operation with the following expressions.

$$F_\alpha : x_{n+1} = \begin{cases} 2x_n, & 0 \le x_n < 2^{a-1} \\ 2\left(I - x_n\right) + 1, & 2^{a-1} \le x_n \le I \end{cases} \quad (3)$$

Where $I = 2^\alpha - 1$, the multiplication (division) operation of Equation. (1) is transformed into the shift operation of Equation. (3) within a finite set of integers. However, since it is still defined in a finite field, the iterative sequence generated using it necessarily enters the periodic state and even some periodic points of small period length appear. For example, when the initial value $x_0 = 1$ and $a = 6$, the iterations are shown in Figure 3, where the values can be seen to cycle in a short period.

In turn, the relevant dynamic parameters are introduced to obtain the dynamic integer tent map. It not only solves the short period problem of integer tent mapping, but also has the property of uniform distribution of tent mapping, as shown in Figure 4. Its specific expression is as follows.

$$F_\alpha : x_{n+1} = \begin{cases} 2g_n, & 0 \le g_n < 2^{\alpha-1} \\ 2\left(I - g_n\right) + 1, & 2^{\alpha-1} \le g_n \le 2^\alpha - 1 \end{cases} \quad (4)$$

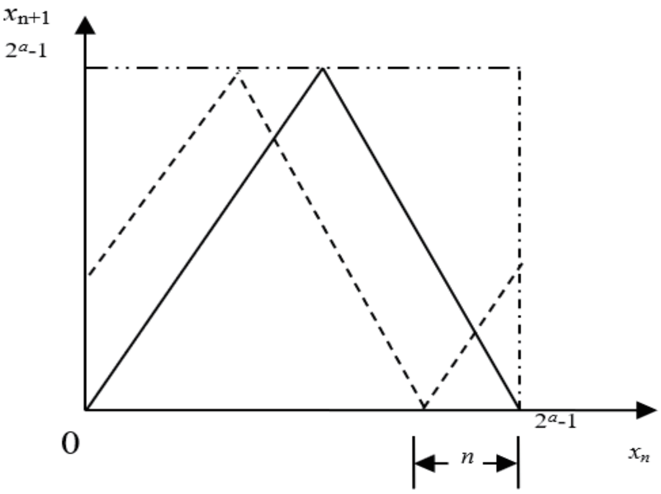


Figure 4: Dynamic integer tent map

Where g_n is taken to be related to the value of and the number of iteration rounds, with the following equation:

$$g_n = \left(x_n + n\right) \bmod 2^\alpha \quad (5)$$

n is the dynamic parameter and also the number of iteration steps, which indicates the distance the tent moves laterally and controls its horizontal movement. x_n indicates the result of the nth iteration step .2^α is the x_n the upper bound of the set of integers taking values, and in this paper, we take the upper bound as 2^{32}. In the iterative operation, as the iteration proceeds, n takes different values to ensure both the dynamics and the stability of the algorithm, which does not make the iteration value change due to the change of dynamic parameters. Again at the initial values $x_0 = 1$ and $\alpha = 6$, the iterations are shown in Figure 5 and it can be seen that the period increases substantially. Liu *et al.* [17]analyzed in detail about the time series generated by dynamic integer tent map, time series frequency distribution, periodicity and correlation, and the experimental results showed that the model has good uniform distribution characteristics, effectively avoids the short-period phenomenon, and its auto-correlation has significant advantages.

The power spectrum analysis is also done for the dynamic integer tent mapping model, and the results are shown in Figure 7. Comparing the power spectrum results without dynamic integer tent mapping, as shown in Figure 6, it can be seen that the overall dense and irregular fluctuation state of the model power spectrum image after adding dynamic parameters indicates that the dynamic integer tent mapping has more complex dynamical behavior.

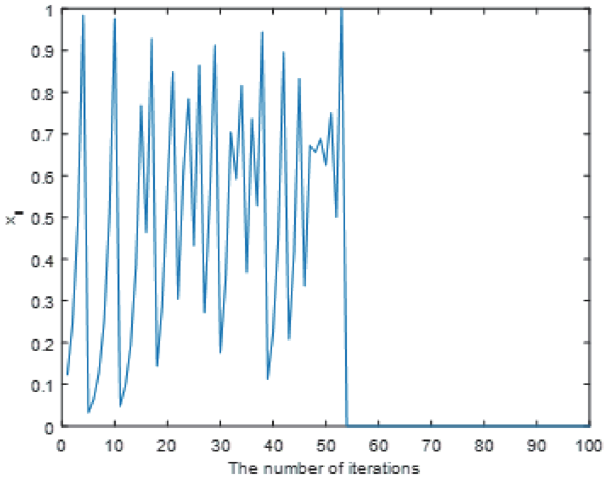


Figure 5: Dynamic integer tent map iteration

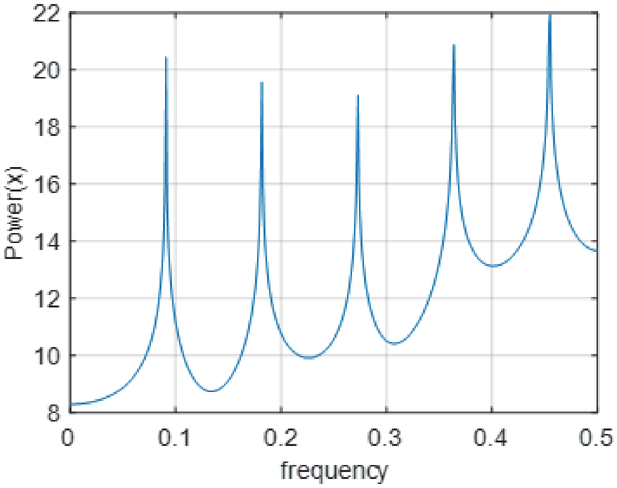


Figure 6: Integer tent mapping power spectrum

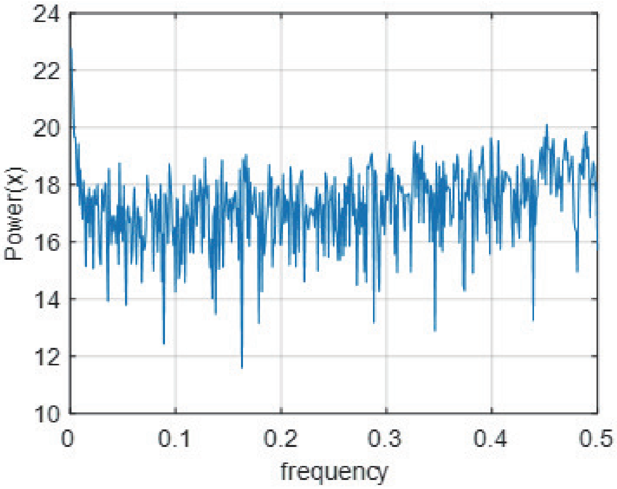


Figure 7: Dynamic integer tent mapping power spectrum

2.2 Two-dimensional Coupled Image Lattice Model

One of the most popular models currently used in the study of space-time chaotic systems is the coupled image lattice (CML). This model is highly valued in research work on space-time chaos due to its high efficiency in numerical diffusion and its more direct use of results from known chaos theories, and it is becoming an important branch in the field of nonlinear dynamics research.The classical one-dimensional one-way and two-way coupled lattice models take the following form.

$$x_{(n+1)}(i) = (1-\varepsilon)f\left(x_n(i)\right)+\varepsilon\left[f\left(x_{(n)}(i-1)\right] \quad (6)$$

$$\begin{aligned} x_{(n+1)}(i) &= (1-\varepsilon)f(x_n(i))+\varepsilon/2[f(x_{(n)}(i-1) \\ &\quad +f(x_{(n)}(i+1)] \end{aligned} \quad (7)$$

Where n is the number of iteration steps; i=1,2,..., L is the grid point coordinates, L is the system size; ε is the coupling coefficient, and its value needs to satisfy $0 \leqq \varepsilon \leqq 1$; $f()$ is the nonlinear map function; the initial value of grid point is a random number within [0,1]. Liu *et al.* [17]compared in detail the initial values and sensitivities of the one-dimensional one-way coupled image lattice model with the one-dimensional two-way coupled image lattice model, and the errors generated by applying a very small order of magnitude perturbation to the initial lattice points to observe the iterative sequence at the same number of lattice points, and concluded that the two-way coupled image lattice model is much better than the one-way .Shang *et al.* [20] based on the one-dimensional two-way coupled image lattice model,constructed an integer coupled image lattice model from the perspective of cryptographic requirements ,its specific expression is as follows:

$$\begin{aligned} x_{(n+1)}(i) = \big[f\left(x_n(i)\right)+f\left(x_{(n)}(i-1)\right. \\ \left.+f\left(x_{(n)}(i+1)\right)\big] \bmod 2^{\alpha} \end{aligned} \quad (8)$$

Where mod denotes the modulo operation, α denotes the number of bits of the system, and 2^{α} denotes the maximum state value that the system can accommodate. The integer model eliminates the coupling coefficients, making each grid point equally affect the next iteration, while avoiding the occurrence of fractions and realizing all integer operations. The simulation results show that the model has good performance in terms of mutual information, sequence complexity, distribution characteristics, difference characteristics and randomness,and can be implemented quickly in a computer.

In this paper, the dimensional is extended from one-dimensional to two-dimensional, and a lattice point can be diffused in four directions in space, and with the evolution of time iteration, each lattice point will affect all other lattice points, and the confusion and diffusion degree of the model is greatly improved. Compared with the one-dimensional two-way coupled image lattice model, the two-dimensional coupled image lattice model has a larger increase in diffusion speed by an order of magnitude from

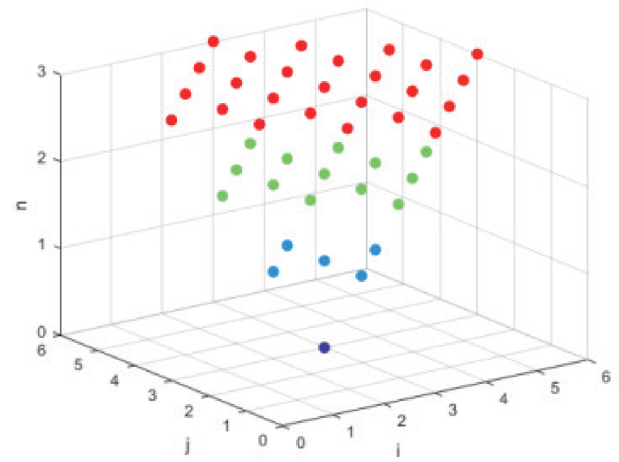


Figure 8: Diffusion process

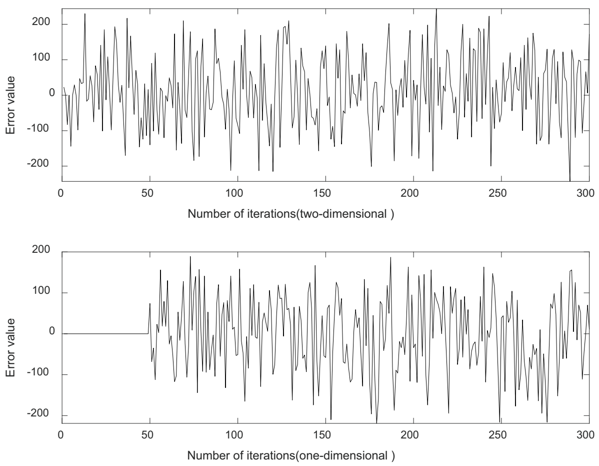


Figure 9: Diffusion process

$2n$+1 to $2n^2$+$2n$+1, although the computational effort increases by a factor of two. Using the dynamic integer tent mapping as a nonlinear function of the two-dimensional coupled image lattice system, its specific expression is as follows:

$$\begin{gathered} x_{(n+1)}(i,j) = \left[f\left(x_{(n)}(i-1,j)+f\left(x_{(n)}(i+1,j)+\right.\right.\right. \\ \left.f\left(x_{(n)}(i,j-1)+f\left(x_{(n)}(i,j+1)\right]\right.\right. \bmod 2^{32} \end{gathered} \tag{9}$$

Where n is the number of discrete time steps, i,j = 1,2,...,L are discrete lattice points, and the boundary conditions:$x_(n)(0,j) = x(n)(L,j), x_(n)(L+1,j) = x(n)(1,j), x_(n)(i,0) = x(n)(i,L), x_(n)(i,L+1) = x_(n)(i,1).(i,j)$ denotes the lattice coordinates of the two-dimensional planar lattice, which contains not only the local reaction processes of the two independent components i and j, but also the diffusion reaction processes of these two independent components on all lattice points of the system of length L. Figure 8 shows the diffusion process of two-dimensional coupled image lattice points at L=5. The figure shows the information diffusion of lattice point $x_(n)(i,j)$ in Equation. (9) after three iterations, and it can be seen that the information of each lattice point diffuses to all lattice points after about four iterations for a summary point of 25, which is much faster than the number of lattice points of the same scale in one dimension.

At the same time, under the same lattice points and the same accuracy, the one-dimensional integer-coupled image lattice model and the two-dimensional integer-coupled image lattice model are tested for perturbation, with the conditions of 100 lattice points, 2 accuracy, and 300 iterations, and the perturbation is 1 applied to lattice point x(L'/2) and lattice point x(L/2,L/2), respectively, and the initial number of steps at which the model starts to appear significantly chaotic is observed, and the results are shown in Figure 9. It can be seen that the two-dimensional integer coupled-image lattice model appears significantly chaotic at the initial iteration. This indicates that its sensitivity is much better than that of the one-dimensional integer-coupled image lattice model.

3 Spark Big Data Platform

Apache Spark was created to improve the efficiency of MapReduce in Hadoop with solving problems such as storage. Spark offers unmatched scalability and is an efficient Swiss Army knife in data processing, providing SQL access, streaming data processing, graph computation, NoSQL processing, machine learning, and more. As an alternative to MapReduce, Spark implements a distributed and fault-tolerant in-memory structure called Resilient Distributed Dataset (RDD), where each RDD is divided into multiple partitions that run on different nodes, enabling parallel processing. RDD can contain objects of any type in Python, Java, Scala, or even user-defined objects. Figure 10 shows the application components and connection interactions in a Spark standalone cluster.

Each component has a specific role to play during the execution of a Spark application. Some components (e.g., clients) are less active during execution, while others are more active during execution, such as components that execute computational functions. The components of a Spark application include a driver, a master, a cluster manager, and at least one executor at the first level. The executor runs on the worker node (worker). All Spark components, including the driver process, master process, and at least one executor process, run on the Java Virtual Machine (JVM), a cross-platform runtime environment engine that executes instructions compiled to Java bytecode.

The core Spark principle is that when a SparkApp application is committed, a SparkContext (App context, which controls the entire lifecycle) object is created by the Driver (running on the master, Task control node) respon-

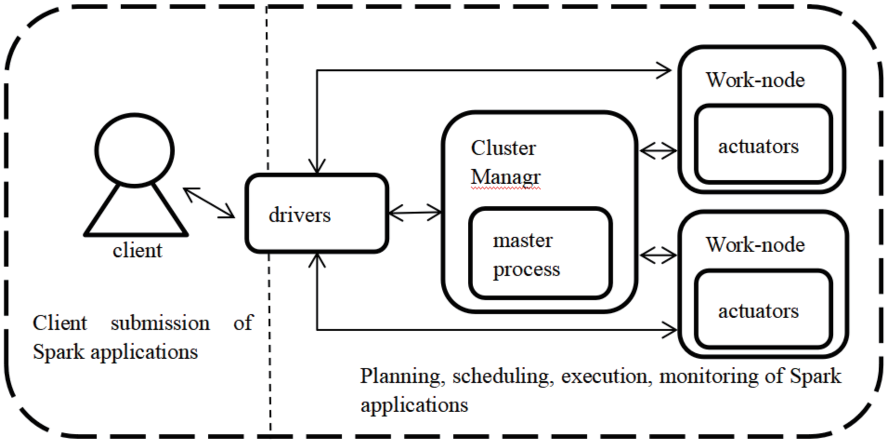

Work-node
actuators
Cluster Managr
master process
drivers
client
Work-node
actuators
Client submission of Spark applications
Planning, scheduling, execution, monitoring of Spark applications

Figure 10: Application components in a Spark standalone cluster

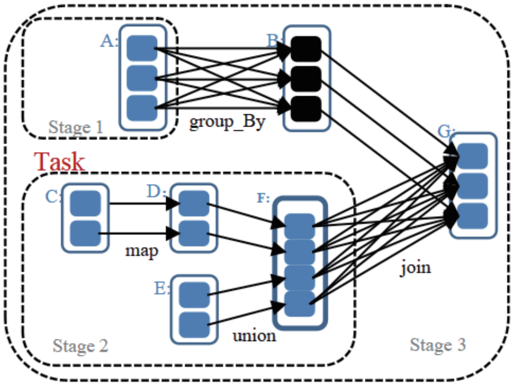


Figure 11: Directed acyclic graph for Spark jobs

sible for communicating with the cluster manager and for resource requests, task allocation and monitoring, etc., to build the basic runtime environment. SparkContext is constructed as a DAG (directed acyclic graph) based on the RDD object dependencies of the user code (e.g. RDD1.jion ...) as in Figure 11, where the solid rounded boxes represent RDDs, the filled matrix represents partitions, and the dashed boxes are Stages, which are handed over to the DAG scheduler, which divides the RDD's DAG into DAGs of individual Stages to form TaskSet (task set), which is then submitted to TaskScheduler (task scheduler),which submits the Task to the Executor (Task execution process) on each Worker (Task work node) to execute the specific Task [23].

4 Hash Algorithm Based on Spark Big Data Platform

Combine the Spark big data platform operation with the Hash function merkle tree type design, first fill the plaintext byte length to the power of 4 (at least 256 bytes), then create RDD, divided plain-text into 4 pieces, each piece is executed separately, in parallel for hash operation. Hash function operation mode for merkle tree structure, four nodes, each node 16 bytes. Every 4 bytes are converted into a 32-bit integer, a total of 16 integers as a two-dimensional coupled image lattice of 16 grids for iterative operations. Finally get each piece of data finally form a new node value, each 128bit, to form a new RDD, and then perform a final Hash operation to get 128bit hash value from it. The flow is shown in Figure 12.

The Hash function is specified as follows.

Notation: $+$: addition operation of mod 2^{32}, $\sim$: bit-by-bit logical inverse, $\oplus$: bit-by-bit logical is-or, $\ll$: left-shift operation, $<<<$: cyclic left-shift operation, $>>>$: cyclic right-shift operation. Define $D = 2^{31}$, the dynamic integer tent map (Equation (4)) $F : x_n \longrightarrow x_n + 1$ in this integer range can be used with the ternary operator (?:) in C language described as:

$$x_n + 1 = x_n < \mathrm{D}?g_n << 1 : (\sim g_n << 1) + 1 \qquad (10)$$

It follows that dynamic integer tent mapping can be implemented with simple logical judgments, logical inversions and shift operations. If implemented in assembly language or hardware, the operation can be further simplified as follows: test whether the highest bit of the word is 0, if it is 0, shift it left, otherwise, each bit is inverted, shifted left, and then added by 1. In the following Hash algorithm design, the result of the logical determination of the dynamic integer tent will also cause the parameter term of dynamic changes, which are uniformly described in the text by the ternary operator (?:) is described in the text. The general procedure of the compression function is given below.

Step 1. Converts the obtained plaintext data to 32-bit integers. Every 4 bytes of plaintext is converted to 1 32-bit integer, forming a total of 16 32-bit integers, m_n, $n = 0, ..., 15$.

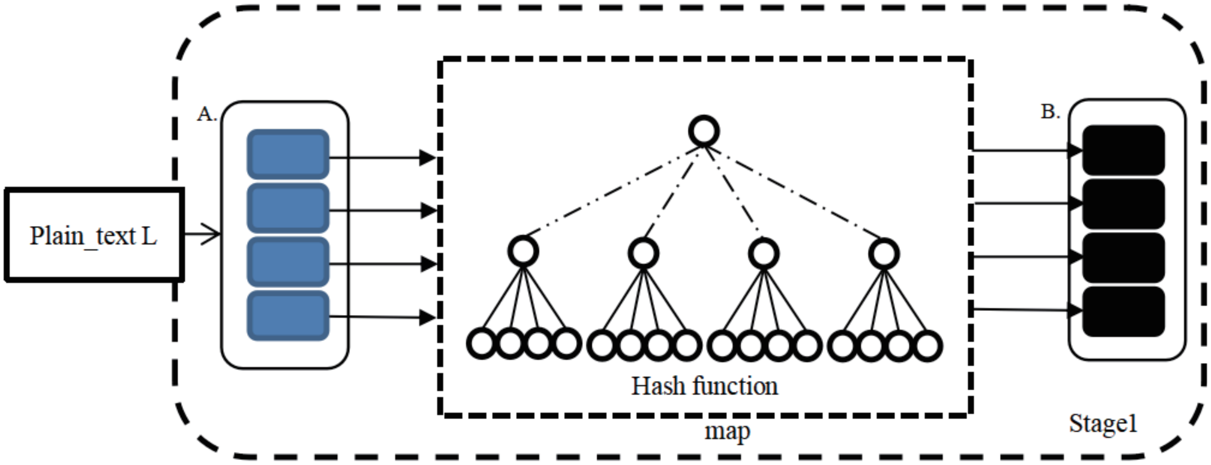


Figure 12: Hash algorithm flow based on Spark platform

Step 2. 16 initial vectors $x_(0)(i,j)$of each 32-bit grid point, $i,j=1,...,4$.

0x01234567UL;0x89abcdefUL;0x3210fedcUL;
0xba987654UL;0x02468aceUL;0x76543210UL;
0xfedcba98UL;0xcdef0123UL;0x456789abUL;

0xeca865420UL;0x083b07fcUL;0x192a26edUL;
0x3a1945deUL;0x3b0864cfUL;0x6ed6c102UL;

0x7fc4e083UL;

Step 3. 16 initial values of 32-bit dynamic parameters $ki^(0), i=0,...,15$.

0x5a827999UL;0x6ed9eba1UL;0x8f1bbcdcUL;

0xca62c1d6UL;0x5793c62aUL;0x66a6b778UL;

0x707b448dUL;0xb7df971UL;0x57d0f4dbUL;

0x4bde569UL;0x6fbb8051UL;0x2f41e8a8UL;

0x2f747580UL;0xca62c1d6UL;0x70f62baeUL;

0x4ada59cfUL;

Step 4. Embedded message. $x_{(0)}(i,j) = m_n + x_{(0)}(i,j), i,j=1,...,4, n=0,...,15$

Step 5. A total of r iteration operations are performed with $r=40+\lfloor d/4\rfloor$ and d is the number of output bits. Each iteration operation is first dynamically transformed so that the lattice values are combined with the number of iteration steps, then mapped transformed using Equation 4, and finally diffusion iterations are performed using the coupled image system model given in Equation 6. $g_{(n)}(i,j) = x_{(n)}(i,j)+n$, n is the number of iterations, i,j= 1,...,4 $n=$ 0,...,15

$x_{(n)}(i,j) = g_{(n)}(i,j) < \mathrm{D}?g_{(n)}(i,j) << 1 : (k_n = k_n >>> 1,$
$\big(\sim g_{(n)}(i,j) << 1\big) + 1\big)$

$x_{(n+1)}(i,j) = \big(x_{(n)}(i-1,j) <<< 4\big) + \big(x_{(n)}(i+,j) <<< 8\big) + \big(x_{(n)}$
$(i,j-1) <<< 12) + \big(x_{(n)}(i,j+1) <<< 16\big) + k_n$

Step 6. Add $x_{(0)}(i,j)$into new value of $x(i,j)$:

$x(i,j) = x(i,j) + x_{(0)}(i,j), i,j$=1,...,4.

Add $ki(0)$ into new value of ki:

$ki = ki + ki^{(0)}, i$=0,...,15

Step 7. After the iteration is completed take the four laterally adjacent lattice points and dissociate and concatenate them to obtain 128 bits of data as a node in the next compression function or as the final output:

$(x(1,1) \oplus x(1,2) \oplus x(1,3) \oplus x(1,4))\|$
$(x(2,1) \oplus x(2,2) \oplus x(2,3) \oplus x(2,4))\|$
$(x(3,1) \oplus x(3,2) \oplus x(3,3) \oplus x(3,4))\|$
$(x(4,1) \oplus x(4,2) \oplus x(4,3) \oplus x(4,4))$.

5 Experimental Results

5.1 Sensitivity Test

In order to test the sensitivity of Hash arithmetic to the initial text, minor changes are made to the initial text and the obtained Hash values are compared and analyzed. Randomly select a text plaintext "Development of Computer Hardware The hardware of digital computers has undergone a series of Development of Computer Hardware The hardware of digital computers has undergone a series of revolutionary changes. The gain in the working speed and function has been impressive. The first transistor was invented in the Bell laboratory in 1948.". Simulation experiments are performed in the following 5 different cases. The output length is 128 bits.

Condition 1 , initial plaintext message.

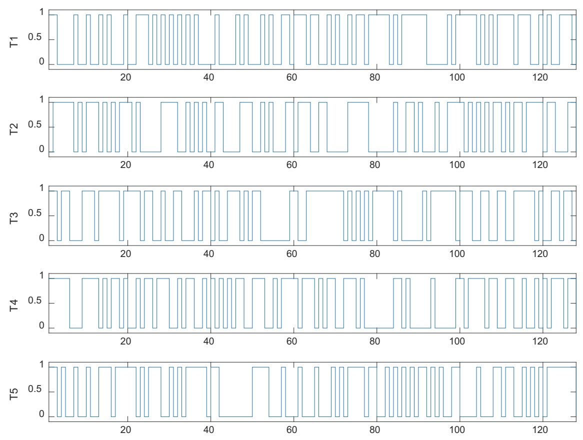


Figure 13: Sensitivity test

Condition 2 , change the first C in the initial text to a B.

Condition 3 , change the first "functions" in the initial text to "function".

Condition 4 , change the full stop at the end of the text to a comma.

Condition 5 , add a space at the end of the text.

The Hash value in hexadecimal representation generated using the algorithm proposed in this paper is as follows, and the resultant output in binary representation is shown in Figure 13, which shows that a small change in the plaintext can have a significant impact on the Hash value.

Hash value at condition 1:
c24a2755 508514dd 94b7d7e0 beaf729d.

Hash value at condition 2:
7d75741e 54c3946e 60f81344 f55adec6.

Hash value at condition 3:
d8efbd93 169d6033 feabe82f dcccfa6c.

Hash value at condition 4:
f8f5a5b9 655672f7 59d01a08 37b32576.

Hash value at condition 5:
d24efae5 7d807888 8af6556a 611b2afe.

5.2 Statistical Analysis of Nonlinear Diffusion Properties

The degree of obfuscation and diffusion of a cryptographic algorithm can be given a probabilistic conclusion by the statistical detection of nonlinear diffusion properties [16]. The analysis of the degree of nonlinear diffusion of cryptographic algorithms by statistical methods usually has to include the aspects of algorithm completeness, avalanche effect and strict avalanche criterion, which are defined as follows: completeness means that each bit of the function output value is related to all bits of the message input, avalanche effect means that a change of any one bit in the message input should cause a change of the output by an average of half a bit, and strict avalanche criterion means that that a change in any one bit of the message input causes a change in each bit of the output with probability 1/2 [8] [21] [5].

Let H be an n-bit input m-bit output Hash algorithm with input vector $x = (x_1, ..., x_n) \in (0,1)^n$ and input vector $x^{(i)} \in (0,1)^n$after changing only the i-th bit of x. Their corresponding output vectors after compressed mapping are denoted as H(x), H$(x^{(i)}) \in (0,1)^m$, respectively.

$(.)j$ denotes the j-th bit of the vector, $w(.)$ denotes the Hamming weight of the vector, and $\#\{.\}$ denotes the potential of the set. Let X be the sample space of the input to the Hash algorithm and note that $a_{ij} = \#\{x \in X | (H(x))j \neq (H(x^{(i)}))j\}$ (where $i = 1, 2, ..., n; j = 1, 2, ..., m$) denotes the number of j-th bit differences between the input vector x and the output vector corresponding to $x^{(i)}$ in X. $b_{ij} = \{\{x \in X | w(H(x(i)) - H(x)) = j\}$(where $i = 1, 2, ..., n; j = 1, 2, ..., m$) denotes the number of differential Hamming weights of j between the input vector x and the output vector corresponding to $x^{(i)}$ in X. Define three statistical measures:

Measure of completeness:

$$d_c = 1 - \frac{\#\{(i,j) \mid a_{ij} = 0\}}{nm} \tag{11}$$

Measure of the extent of the avalanche effect:

$$d_a = 1 - \frac{\sum_{i=1}^{n} \left| \frac{1}{\#X} \sum_{j=1}^{m} 2jb_{ij} - m \right|}{nm} \tag{12}$$

Metric for the degree of rigorous avalanche:

$$d_{sa} = 1 - \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} \left| \frac{2a_{ij}}{\#X} 1 \right|}{nm} \tag{13}$$

If $H(.)$ is a random transformation and $\mathrm{Z}_{\alpha/2}$ denotes the $\alpha/2$ quantile of the standard normal distribution, the paper Zhu *et al.* [28] gives the following conclusion. The sample size X for the test should be at least $nm \times (\mathrm{Z}_{\alpha/2})^2$;

$p(d_c) = 1 - 2^{-\#X} \approx 1.0$;

$\mathrm{E}\{d_a\} = 1.0 - \sqrt{2/(\pi \times m \times \#X)}$, with a confidence interval of $\left(\mathrm{E}\{d_a\} - \mathrm{Z}_{\alpha/2}\sqrt{1/(n \times m \times \#X)}, \mathrm{E}\{d_a\} + \mathrm{Z}_{\alpha/2}\sqrt{1/(n \times m \times \#X)}\right)$ $\mathrm{E}\{d_{sa}\} = 1.0 - \sqrt{2/(\pi \times \#X)}$,with a confidence interval of $\left(\mathrm{E}\{d_{sa}\} - \mathrm{Z}_{\alpha/2}\sqrt{1/(n \times m \times \#X)}, \mathrm{E}\{d_{sa}\} + \mathrm{Z}_{\alpha/2}\sqrt{1/(n \times m \times \#X)}\right)$.

It is important to note here that the ideal Hash function should be a random mapping from all possible input values to a finite set of possible output values. Strictly speaking, there is no such Hash function as a random mapping. This is because Hash functions are deterministic, and the deterministic and uniform output property implies that the entropy of the output is greater than the entropy of its input. However, according to Shannon's entropy theory, a deterministic function can never amplify the entropy. Our design goal is to make the Hash function indistinguishable from a random mapping in terms of probability distribution. An actual Hash function that tests its statistics d_c, d_a, d_{sa} and falls into its confidence interval indicates that the Hash algorithm satisfies the basic requirement of nonlinear diffusion, i.e., the Hash function can be considered to have good completeness and avalanche effect, satisfying the strict avalanche criterion [17].

Taking the input length n=512 bits and the output length m=128 bits, the following results are obtained theoretically at the significance level α=0.05:

$Z_{\alpha/2}$=1.96, picking a sample size X of 260,000;

$d_c = 1.000000$;

$E\{d_a\}$=0.999888, with a confidence interval of (0.999860, 0.999863);

$E\{d_s a\}$=0.998589, with a confidence interval of (0.998434, 0.998437).

An example of one-column is shown as Table 1.

Table 1: Beat-by-beat statistics of the diffusion performance of the algorithm in this paper

Number of iterative beats	dc	da	dsa
1	1.000000	0.999861	0.998437
2	1.000000	0.999867	0.998441
3	1.000000	0.999870	0.998437
4	1.000000	0.999869	0.998443
5	1.000000	0.999866	0.998444
6	1.000000	0.999858	0.998435
7	1.000000	0.999866	0.998436
8	1.000000	0.999862	0.998432
9	1.000000	0.999869	0.998434
10	1.000000	0.999863	0.998427

We randomly select a sample set X of 260,000 sets of 512 bit words as the message input for the Hash algorithm in this paper under the above conditions, and the actual test results are shown in Table 1. As can be seen from Table 1, the statistics dc, da, and dsa fall into their respective confidence intervals after 1 beat of iteration, and all subsequent iterations also fall roughly into or close to their respective confidence intervals at the significance level $\alpha = 0.05$, indicating that the algorithm in this paper satisfies the basic requirement of nonlinear diffusivity of the Hash algorithm.

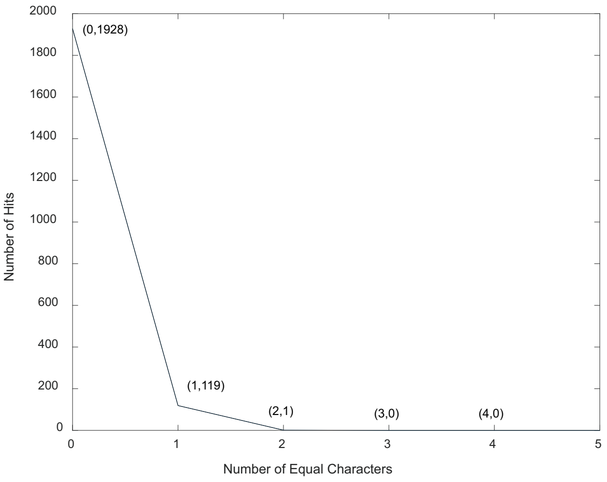


Figure 14: Sensitivity test

5.3 Collision Analysis and Birthday Attack

The collision resistance of a Hash function is an important indicator to test its security, if two different inputs X1 ≠ X2 can be found and their Hash values H1 = H2, then it becomes a collision. While the birthday attack is essentially similar to the collision problem, it deals with the problem of the probability that two random inputs have the same Hash value obtained after being processed by the Hash function. For this algorithm, in the two-dimensional coupled image lattice model, the message is embedded in the corresponding lattice before starting the iteration, and the dynamic integer tent mapping is used as a nonlinear function of the model, which increases the chaos and diffusion when processing the data, and the adjacent lattice points receive mutual influence, and the diffusion of the whole model increases by one level with each further iteration, and eventually all the lattice points will be related to the initial lattice and the embedded message value, so that you have me and I have you, which will ensure that the final Hash value of any bit is related to all bits of the message, and small changes in the message, amplified by the continuous diffusion of the iterative process, will eventually lead to completely different Hash results.

Here, the collision resistance of the algorithm is tested by the method of reference Wong [22] . That is, a segment of plaintext is randomly selected in the plaintext space to find and store its Hash value in ASCII code form, and then the value of 1 bit in the plaintext is randomly selected and changed to another new Hash result, and the number of characters with the same value of ASCII code in the same position is recorded. The above experiment is repeated 2048 times and the results obtained are shown in Figure 14 . It can be seen from the figure that the maximum number of identical characters is only 2. The degree of collision is very low.

Meanwhile, character distance is a statistic used to test whether the Hash values generated by two different plaintext messages are independent of each other, defined as follows.

$$d = \frac{1}{s}\sum_{i=1}^{s} |H_1[i] - H_2[i]| \tag{14}$$

Where d is the average character distance, H1 and H2 are the values corresponding to the i-th ASCII character in the two Hash values, respectively, and s is the byte length of the resulting Hash value. If the two tested Hash values are independent and obey uniform distribution, the average character distance obtained should be 85.33 in theory. The test result of the algorithm in this paper is 85.19, which is closer to the theoretical value, and it can be considered that the Hash values obtained by this algorithm are independent and obey uniform distribution.

Table 2 gives the comparison of the absolute difference between the number of ASCII characters at the same position and the 128-bit hash value of the algorithm in this paper and the selected literature algorithm based on N=2048 random tests. The results show that the algorithm proposed in this paper has strong confusion and diffusion ability and strong collision resistance.

Table 2: Beat-by-beat statistics of the diffusion performance of the algorithm in this paper

Algorithm	Number of collisions				d
	0	1	2	3	
this paper	1928	119	1	0	85.19
Li et al [14]	1930	117	2	0	84.38
Nabil et al [18]	1931	114	3	0	80.72
Nabil et al [19]	1806	229	13	0	84.85
Lin et al [15]	1931	114	3	0	90.23

5.4 NIST Randomness Test

The NIST Test Suite is a statistical package consisting of 15 tests, these are designed to test random (arbitrary length) binary sequences generated by hardware or software based cryptographic random or pseudo-random number generators. The main process of the test is to observe the magnitude of the obtained P_value compared with the set value α to determine whether it passes or not, if P_value $\geq \alpha$, it can be considered to pass the test, and it is necessary to analyze the pass rate of P_value, i.e., the ratio of all sequences passing the test purpose to all test sequences, if the pass rate reaches 0.96 or more, it means that the sequence meets the randomness requirement. Using the algorithm proposed in this paper, 100 binary sequences of 106 bits are generated and tested, and the P_value and pass rate in the results are analyzed. The test results are shown in Table 3. The P_values of all items meet the requirements, and the pass rates are above 0.96, which indicates that the sequence values generated by the algorithm meet the randomness requirements.

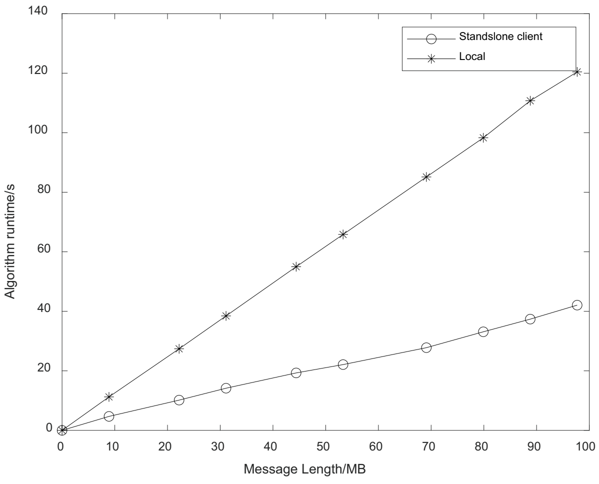


Figure 15: Comparison of execution times

5.5 Implementation Efficiency Tests

The physical machines used for the experiments are Intel(R) Core(TM) i3-2120 CPU @ 3.30GHz, with four nodes, each with 1GB of memory, and the number of processors set to 1. The operating system version installed on each node is 64-bit Linux Ubuntu 16.04.6, and the JDK used is JDK1.8. Scala version is Scala 2.11.11, Hadoop version is Hadoop 2.7.3, and Spark version is Spark 2.2.0.

For the algorithm using Python programming implementation, Spark uses Standalone client mode to call the Python program after starting, compared to Standalone cluster mode, Standalone client mode runs the task process driver on the client side, which facilitates the relevant operations on the program.

The text files of different sizes are processed in Spark cluster and the time consumed for the processing is recorded, while the same size files are processed in a local environment and the time consumed is recorded. The time consumed for image encryption processing in the two different cases is compared and the results of the comparison are shown in Figure 15. It can be seen that the algorithm runs significantly faster in the clustered environment than in the local mode.

6 Conclusions

This paper is based on Spark big data platform with dynamic integer tent mapping and two-dimensional coupled image lattice system with additional dynamic parameters added to increase the chaotic properties. The main elements include integerizing the tent mapping and making it dynamic to further increase the chaotic characteristics;

Table 3: NIST random test results

test	P_value	Proportion
Frequency	0.708404	1.00
Block Frequency	0.753041	1.00
Cumulative sums	Success	1.00
Runs	0.370824	0.97
Longest Run of Ones	0.942695	1.00
Rank	0.394923	0.99
Discrete Fourier Transform	0.310574	0.98
Non-Overlapping Template Matchings	Success	0.99
Overlapping Template Matchings	0.153157	1.00
Universal Statistical	0.675325	0.99
Approximate Entropy	0.314443	1.00
Random Excursions	Success	0.98
Random Excursions Variant	Success	0.99
Serial	Success	1.00
Linear Complexity	Success	1.00

combining the two-dimensional coupled image lattice system, adding dynamic parameters, and using the dynamic tent mapping to control the change of dynamic parameters, and using the parallelism feature of Spark platform to make the algorithm execution efficiency nearly doubled compared to the stand-alone operation. It can be said that the dual dynamic parameters enhance the confusion and diffusion nature of the system to a large extent. Several tests have shown that the algorithm has strong obfuscation and diffusion capabilities and high execution efficiency, which can be applied to a variety of cryptographic situations.

References

[1] A. Akhavan, A. Samsudin, and A. Akhshani, "A novel parallel hash function based on 3d chaotic map," *EURASIP Journal on Advances in Signal Processing*, vol. 2013, no. 1, pp. 1–12, 2013.

[2] T. Y. Chang, M. S. Hwang, W. P. Yang, "A new multi-stage secret sharing scheme using one-way function", *ACM SIGOPS Operating Systems Review*, vol. 39. no. 1, pp. 48-55, 2005.

[3] T. Y. Chang, M. S. Hwang, W. P. Yang, "Cryptanalysis of the Tseng-Jan anonymous conference key distribution system without using a one-way hash function", *Information & Security: An International Journal*, vol. 15, no. 1, pp. 110-114, 2004.

[4] X. Cui, P. Zhu, X. Yang, K. Li, and C. Ji, "Optimized big data k-means clustering using mapreduce," *The Journal of Supercomputing*, vol. 70, no. 3, pp. 1249–1259, 2014.

[5] H. Feistel, "Cryptography and computer privacy," *Scientific American*, vol. 228, no. 5, pp. 15–23, 1978.

[6] D. Harnie, M. Saey, A. E. Vapirev, J. K. Wegner, A. Gedich, M. Steijaert, H. Ceulemans, R. Wuyts, and W. De Meuter, "Scaling machine learning for target prediction in drug discovery using apache spark," *Future Generation Computer Systems*, vol. 67, pp. 409–417, 2017.

[7] J. Ji and Y. Zheng, "A spark-based algorithm for high-dimensional k-nearest neighbor connectivity(in chinese)," *Computer Engineering and Design*, vol. 39, no. 6, pp. 2544–2549, 2018.

[8] J. B. Kam and G. I. Davida, "Structured design of substitution-permutation encryption networks," *IEEE Trans on Computers*, vol. 28, no. 10, p. 747, 1979.

[9] P. Karthik and P. S. Bala, "A new design paradigm for provably secure keyless hash function with subsets and two variables polynomial function," *Journal of King Saud University-Computer and Information Sciences*, 2019.

[10] W. Kim, Y. Kim, and K. Shim, "Parallel computation of k-nearest neighbor joins using mapreduce," in *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016, pp. 696–705.

[11] H. S. Kwok and W. K. S. Tang, "A chaos-based cryptographic hash function for message authentication," *International Journal of Bifurcation and Chaos*, vol. 15, no. 12, pp. 4043–4050, 2005.

[12] W. J. Li and Z. H. Zhou, "Big data hash learning:status and trends(in chinese)," *Science Bulletin*, vol. 60, no. 6, pp. 485–490, 2015.

[13] Y. Li, S. Deng, and D. Xiao, "A novel hash algorithm construction based on chaotic neural network," *Neural Computing and Applications*, vol. 20, no. 1, pp. 133–141, 2011.

[14] Y. T. Li and F. G. Ge, "Cryptographic and parallel hash function based on cross coupled map lattices suitable for multimedia communication security," *Multimedia Tools and Applications*, vol. 78, no. 12, 2019.

[15] Z. Lin, C. Guyeux, S. Yu, and Q. Wang, "On the use of chaotic iterations to design keyed hash function," *Comput*, vol. 22, pp. 905–919, 2017.

[16] J. D. Liu, "Extended integer tent mapping with dynamic hash functions(in chinese)," *Journal of Communication*, vol. 31, no. 05, pp. 51–59, 2010.

[17] J. D. Liu and X. L. Fu, "Construction of spatiotemporal chaotic one-way hash function based on coupled tent mapping(in chinese)," *Journal of Communication*, vol. 06, no. 9, pp. 30–38, 2007.

[18] A. Nabil, E. A. Safwan, M. H. Thang, and D. Olivier, "Design and security analysis of two robust keyed hash functions based on chaotic neural networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, 2019.

[19] A. Nabil, E. A. Safwan, M. H. Thang, and D. Olivier, "Designing two secure keyed hash functions based on sponge construction and the chaotic neural network," *Entropy (Basel, Switzerland)*, vol. 22, no. 9, 2020.

[20] K. Shang, J. D. Liu, X. Zhang, and H. H. Hu, "Integer nonlinear coupled image lattice model and its performance analysis(in chinese)," *Computer Science and Exploration*, vol. 11, no. 03, pp. 389–395, 2017.

[21] A. F. Weister and S. E. Tavares, "On the design of s-boxes. advances in cryptology-crypto," *Berlin:Springer -Verlag*, vol. 85, pp. 523–533, 1986.

[22] K. K. Wong, "A combined chaotic cryptographic and hashing scheme," *Physics Letters A*, vol. 5, p. 307, 2003.

[23] F. Xiao, L. J. Zhang, and Z. Wang, *park big data technologies and applications*, 1st ed., ser. 10. Beijing, P. R. China: Beijing: people's post and telecommunications publishing house, 2 2018, vol. 4, computer and internet.

[24] L. Yin, L. Qin, Z. Jiang, and X. Xu, "A fast parallel attribute reduction algorithm using apache spark," *Knowledge-Based Systems*, vol. 212, p. 106582, 2021.

[25] H. J. Zhai, M. Zhang, T. T. Wang, and P. Hao, "K-nearest neighbor algorithm for large datasets based on hashing technique and mapreduce(in chinese)," *ComputerScience*, vol. 44, no. 5, pp. 210–214, 2017.

[26] B. Zhang and J. J. Le, "Distributed hash joining algorithm for mapreduce based on column storage(in chinese)," *ComputerScience*, vol. 45, no. 5, pp. 471–475+505, 2018.

[27] P. Zhang, X. Zhang, and J. Yu, "A parallel hash function with variable initial values," *Wireless Personal Communications*, vol. 96, no. 2, pp. 2289–2303, 2017.

[28] M. F. Zhu, B. D. Zhang, and S. W. Lv, "A statistical method of blockcipher on diffusion & propagation(in chinese)," *Journal of china institute of communications*, vol. 23, no. 10, pp. 122–128, 2002.

Biography

Jiandong Liu Received his Master degree in New Technology of Electrical Theory from Tianjin University in 1995, and is currently the Dean of School of Information Engineering, Beijing Institute of Petroleum and Chemical Technology, and Professor of School of Information Engineering, Beijing Institute of Petroleum and Chemical Technology. His main research interests include Chaotic Codes and Information Hiding, etc..

Yujie Liu Received his Master degree in chemical engineering from Beijing Institute of Petrochemical Technology in 2022, his research interests include chaotic ciphers, hash functions, etc.

Bo Li Received her Master degree in Control Science and Engineering from Beijing Institute of Petrochemical Technology in 2022, her research interests include chaotic cryptography, image encryption, etc.

Privacy-preserving Broadcast Protocol in Vehicular Ad Hoc Networks

Guang Yang, Leyou Zhang, and Ruonan Ma
(Corresponding author: G. Yang, L. Zhang)

School of Mathematics and Statistics Xidian University, Xi'an, Shaanxi, China
(Email: 982782903@qq.com; lyzhang@mail.xidian.edu.cn)

(Received Oct. 4, 2022; Revised and Accepted Apr. 1, 2023; First Online Apr. 27, 2023)

Abstract

In vehicular ad hoc networks (VANETs), a vehicle always sends and receives different sensitive information on the road. It will issue some threats to the privacy of a vehicle and users since the communication can link their identity to their physical entity. Hence secure broadcast has been an important issue and gained more and more researchers' attention in real life. In addition, authorities and official vehicles should confirm the authenticity of the reported node without exposing the sender's identity. The anonymous broadcast protocol can support the secure and anonymous transmission of vehicle safety broadcast messages, achieving the receivers' privacy protection. This article introduces an attribute-based encryption system's new anonymous broadcast protocol. And a hybrid broadcast algorithm is put forth to show the efficiency and security of the introduced scheme. Furthermore, the sharing information of the proposed scheme achieves constant size, which is essential for the resource-constrained VANETs since it consists of the rapid movement of the nodes. The proposed scheme also supports the dynamic management of the vehicles in the VANETs and does not need to fix the maximum depth of the VANETs. Finally, computational results confirm the merits of the proposed scheme.

Keywords: Anonymous Broadcast Protocol; VANETs; CP-ABE; Privacy Protection

1 Introduction

As an important component of the intelligent transportation system (ITS), vehicular ad hoc networks (VANETs) have gotten much attention and interesting at present [3, 21, 31]. In this network, it is characterized by vehicles supported by fixed infrastructure, where the infrastructure is fixed along road sides and on board communication unit are set on vehicles. Each vehicle is either to Vehicle communication (V2V) or to Infrastructure communications (V2I) [11–13]. The vehicles share their current status, including position, speed, road conditions, and safety information. In addition, the vehicles must record data and send them to the infrastructure where it in turn report them to the service provider (or center). As a special mobile ad hoc networks (MANETs), VANET has its own behaviors and characteristics, such as availability of location information, frequent link disconnections etc [43]. In addition, a VANET inherits all the known and unknown security weaknesses associated with MANETs. All of them issue many security and privacy leak problems. Any behavior of invalid vehicular users, such as a modification and replay attack with respect to the disseminated messages [38], could be fatal to the other vehicular users. In addition, privacy information may be sensitive, such as the driver's name, license plate, speed, position, and traveling routes along with their relationships. Hence a natural problem is how to preserve the privacy while the security is achieved. Additionally, in the Vehicle-to-everything (V2X) communications, a vehicle may gather and process of large numbers of messages (data) packets. If the false or bogus messages are broadcasted it may lead to severe accidents and threaten human lives [3]. Anonymous broadcast protocol can guarantee the data confidentiality and provides the privacy protection for the vehicle. It has been one of hot topics in VANETs at present [14–16].

1.1 Related works

Following [30], the anonymous broadcast can be classified into tree types, infrastructure-based anonymity, cryptography-based anonymity or protocol-based anonymity.

In [41], [36], authors used specialized network servers to conceal a broadcaster's ID from attackers. Beresford and Stajano in [5] introduced the anonymizer concept to keep a user identity from departing the zone to its identity upon entry. Another concept called k-anonymity was introduced in [25], [23] to protect a use's identities from Location-Based Service (LBS) providers. Cryptography-based anonymity is a challenging method at present. A more previous work was given in [44], where it is based on public key infrastructure (PKI). Then a solution is based on group signature [18]. Unfortunately, in some situations such as revocation, it does not work well [30]. Another solution is based on the ring signature and encryption algorithm. In [10], authors proposed anonymous authenticated key agreement protocol based on the standardized Elliptic Curve Integrated Encryption Scheme and on ring signatures. But the computational power and storage requirements was not economical. Protocol-based Anonymity was introduced in [21]. In [40], Sun etal proposed a real-time adaptive dissemination (RTAD) scheme. Then the TRAck DEtection (TRADE) and distance defer transmission (DDT) were proposed in [4]. In 2005, the smart broadcast algorithm (SBA) [22] and contention based dissemination (CBD) [42] protocol and time reservation-based relay node selecting algorithm (TRRS) [28] were issued respectively. The others are Urban multi-hop protocol (UMB) [29], BROADCOMM [20], fast broadcast (FB) protocol [35], and REAR protocol [27]. The detailed introduction is referred to the survey [21].

These schemes have the following shortcomings.

- Do not support dynamic managements of the vehicles.
- There is not a best trade-off between security and privacy preserving.
- The size of delivery messages is large. It is hard to send or collect such information for VANET because of rapid movement of the nodes.

This paper focuses on cryptography-based anonymity with the hybrid technique. Our protocol is based on the attribute based encryption. Compared the previous work, the proposed scheme does not need to distribute certificates and Key Recovery. The proposed schemes achieves short public keys, constant size delivery messages. In the prime order groups, the proposed scheme achieves compact security and anonymity. In addition, the proposed scheme supports the dynamic management. When a new vehicle applies to join in the networks, the center only generates the corresponding the private keys by using the communal parameters instead of resetting the system. And when a vehicle leaves the networks, the center revokes its attribute by modifying its access policy. In addition, the proposed protocol is not fixed the scale number at the beginning of the setup phase.

The rest of this paper is structured as follows. A brief review of system model for the VANET and some basic preliminaries are presented in Section 2. The proposed protocol is described in Section 3. In Section 4, the security and performance analysis of the protocols are discussed. We conclude the paper with a summary in Section 5.

2 Preliminaries

2.1 VANET Architecture

VANETs includes a trusted authority (TA), immobile road side units (RSU) and vehicles equipped with onboard units (Vehicle/OBU) [38].

TA: TA manages the registration of RSUs and Vehicle/OBU.

RSU: It connects the TA and the Internet backbone to support diversified services. It also stores data from the vehicles.

Vehicle/OBU: It mainly communicates with each for sharing local traffic information for making a request to obtain latest updates from other vehicles and incar sensors data.

2.2 Broadcast Protocol

A broadcast protocol consists of four algorithms, System initialization, Vehicle registration (Vehicle Grant and Key Generation), Message Broadcast and Share Message. They are described as follows.

System initialization. Suppose there are some vehicles and Infrastructures to set up a VANET. It will output public keys and Master key.

Vehicle Grant. If a vehicle want to join in the VANETs, it sends the information to the Center. Then it will receive the corresponding private keys which will make it recover the received broadcast messages.

Message Broadcast. If a vehicle makes a V2V or V2I communication, it will run the Encrypt (PK,S) to get the broadcast messages (ciphertexts) and broadcast them to others.

Share Message. After receiving a broadcast message, the vehicle recovers the messages with its private keys.

2.3 Attribute-based Encryption

For the security of data and preserving privacy, the data is stored in the encrypted form. One disadvantage of encrypting data is that it severely limits the ability of users to selectively share their encrypted data at a fine-grained level. Sahai and Waters [37] made some initial steps to solving this problem by introducing fuzzy identity-based encryption, which issued the concept of Attributed-Based Encryption (ABE) [24]. In ABE, ciphertexts and keys are associated with attribute sets and access structure over attributes, where a particular key can decrypt a particular ciphertext only if there is a match between the attributes of the ciphertext and the user's key [19, 32]. There are two kinds of ABE including ciphertext-policy ABE (CP-ABE) [6] and key-policy ABE (KP-ABE) [24].

In KP-ABE, the private key of a user is associated with an access structure that specifies which type of ciphertext the user can decrypt. In a CP-ABE scheme, a user can express any access policy, stating what kind of receivers is able to decrypt the message.

ABE scheme supports fine-grained access control, but an attacker can guess the purpose of the message from the ciphertext by using receiver's attributes. For example, attacker can guess that the receiver is a doctor if some of the attributes are "XX hospital, medicine, treatment cycle, cancer", etc. Therefore, protecting receiver's identity while using ABE is an important problem in the real life. Anonymous ABE (AABE) is issued from anonymous identity-based encryption (IBE). In an anonymous IBE scheme, the ciphertext does not leak the identity of the recipient. The first IBE known to be inherently anonymous was that of Boneh and Franklin [8]. The further application is public key encryption with keyword search (PEKS) scheme [39]. Later, Abdalla *et al.* [1] formalized it. In PEKS, one can search keywords on encrypted data and the capability for search is delegable. AABE inherits the feature of IBE. It is introduced in [1, 17, 39] as a promising public-key primitive that allows sender in achieving receiver anonymity in ABE. There are two kinds of ABE, CP-ABE and KP-ABE. In this paper, we discuss anonymous CP-ABE. In anonymous CP-ABE, access policy is hidden in the ciphertext. A receiver decrypts a ciphertext using secret key belongs to his attributes. If the secret key matches with the access policy, the receiver can recover the ciphertext successfully. Otherwise the receiver cannot get what access policy is specified by the encryptor.

2.4 Access Structure

Our construction will employ AND-gate on multi-valued attributes access structure. The access structure of AND-gate on multi-valued attributes is described as follows.

Let $U = \{att_1, att_2, \cdots, att_n\}$ be a set of attributes. For $att_i \in U$, $S_i = \{v_{i,1}, v_{i,2}, \cdots v_{i,m_i}\}$ is a set of possible values, where m_i is the number of possible values for each att_i. Let $L = [L_1, L_2, \cdots L_n]$ be an attribute list for a user where $L_i \in S_i$. Let $A = [w_1, w_2, \cdots w_n]$ be an access structure where $w_i \in S_i$. The notation $L \models A$ expresses that an attribute list L satisfies an access structure A and $\not\models$ refers to not satisfy symbol. If and only if $L_i = w_i$ (where $i = 1, 2, \cdots, n$), we say that the attribute list L satisfies the access structure $\mathbb{A}$ and show as $L \models \mathbb{A}$. Let $L \not\models \mathbb{A}$ denote that L does not satisfy $\mathbb{A}$.

2.5 Ciphertext-policy Attribute-based Encryption (CP-ABE)

Based on [23-29], CP-ABE is given as follows.

Setup$(1^\lambda, U) \to$ **(*PK, MSK*):** It takes as input a security parameter λ and attribute universe description U. Then output the public key PK and the master secret key MSK.

KeyGen(*MSK, PK, S*) $\to$ ***SK*:** For any a set of attributes S, it uses MSK and PK to generate a secret key SK.

Encrypt$(PK, M, \mathbb{A}) \to$ ***CT*:** It takes in PK, the message M, and the access structure $\mathbb{A}$, and then outputs the ciphertext CT.

Decrypt(*PK,CT, SK*) $\to M$**:** It takes as input PK, CT and SK. If the match with $\mathbb{A}$ holds, it recovers M.

2.6 Security Model

The security of the proposed scheme is based on the anonymous ABE (AABE) [28,29]. It is recalled as follows.

***Init*:** The attribute V^* and two policy W_0^*, W_1^* are outputted as challenged issues.

***Setup*:** The simulator generates the AABE protocol and outputs PK to the adversary but keeps the master key secret.

***Phase 1*:** Adversary adaptively makes queries $q_1, \cdots, q_m$. Each is run as :

- Extract query. Output a target V_i with $V_i \neq V^*$ and V_i can not match W^*. The simulator runs *KeyGen* to obtain d_i as a respond to the query. Then sends it to the adversary.

***Challenge*:** The adversary outputs two messages M_0, M_1 (with the same size). The challenger selects a random bit $b \in \{0, 1\}$ and generates the challenge ciphertext $C = Encrypt(params,\ V^*, M_b, W_b^*)$, as the respond to the adversary.

***Phase 2*:** Run as Phase 1.

***Guess*:** Finally, the adversary outputs a $b' \in \{0, 1\}$ as its guess. If $b = b'$, the adversary will win this game.

2.7 Decisional Bilinear Diffie-Hellman Exponent (BDHE) Assumption

DBHE problem [33] is defined as follows: Given as input a random tuple $(g, y_0, y_1, \cdots, y_n, y_{n+2}, \cdots, y_{2n+2})$, where $y_i = g^{\alpha^i}, y_0 = g^c$ and $\alpha, c \in Z_p^*$, output $e(g, y_0)^{\alpha^{n+1}}$ or $e(g, g)^{\alpha^{n+1}c}$. Let $TU = (g, y_0, y_1, \cdots, y_n, y_{n+2}, \cdots, y_{2n+2})$. An algorithm $\mathcal{A}$ that outputs $e(g, y_0)^{\alpha^{n+1}}$ has advantage ε in solving it if

$$\Pr(A(TU) = e(g, y_0)^{\alpha^{n+1}}) \geq \varepsilon.$$

The decisional BDHE problem in G is defined in the usual manner. Given as input a random tuple $(g, y_0, y_1, \cdots, y_n, y_{n+2}, \cdots, y_{2n+2}, T)$, Algorithm B's goal is to output 1 when $T = e(g, y_0)^{\alpha^{n+1}} = e(g, g)^{\alpha^{n+1}c}$ or 0 otherwise. Let $TU = (g, y_0, y_1, \cdots, y_n, y_{n+2}, \cdots,\ y_{2n+2})$.

Algorithm B that outputs $b \in \{0,1\}$ has advantage ε in solving decisional BDHE in G if

$$|\Pr(B(TU, e(g, y_0)^{\alpha^{n+1}}) = 0) - \Pr(B(TU, T) = 0)| \geq \varepsilon.$$

We call a (t, ε)-Decisional BDHE Assumption holds, if the advantage ε of t-time algorithm is negligible advantage in solving decisional BDHE game.

2.8 Decisional Linear Assumption (LA)

[7] defines the decisional LA as: Given a tuple

$$(g, g^{z_1}, g^{z_2}, g^{z_1 z_3}, g^{z_2 z_4}, T),$$

decide whether $Y = g^{z_3+z_4}$.

Some modified versions (called Augmented Decisional linear problem (ADL)) [34] are given as follows.

1) Given a tuple

$$(g, g^{z_1}, g^{z_2}, g^{z_2^2}, \cdots, g^{z_2^{n+1}}, g^{z_2^n/z_1}, g^{z_2^{n+1}/z_3}, g^{z_4}, Y),$$

decide whether $Y = g^{z_1(z_3+z_4)}$.

2) Given a tuple

$$(g, g^{z_1}, g^{z_2}, g^{z_2^2}, \cdots, g^{z_2^n}, g^{z_2^{n+2}}, \cdots, g^{z_2^{2n}}, g^{z_3}, g^{z_4}, g^{z_4 z_2},$$
$$\cdots, g^{z_2^n z_4}, Y),$$

decide whether $Y = g^{z_1(z_3+z_4)}$.

3) Given a tuple

$$(g, g^{z_1}, g^{z_2}, g^{z_2^2}, \cdots, g^{z_2^n}, g^{z_2^{n+2}}, \cdots, g^{z_2^{2n}}, g^{z_3}, g^{z_4}, g^{z_4 z_2},$$

$g^{z_2^n z_4}, g^{z_1 z_2}, \cdots, g^{z_2^n z_1}, Y)$, decide whether $Y = g^{z_1(z_3+z_4)}$.

2.9 Symmetric Bilinear Groups

We briefly review bilinear maps and bilinear map groups. We use the following notation:

- G and G_1 are two (multiplicative) cyclic groups of prime order p;
- g is a generator of G;
- e is a bilinear map $e : G \times G \to G_1$.

The bilinear map e has the following properties:

1) Bilinearity: for all $u, v \in G, a, b \in Z_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$;

2) Non-degeneracy: $e(g, g) \neq 1$;

3) Computability: There is an efficient algorithm to compute $e(u, v)$ for all $u, v \in G$.

3 The Proposed Protocol

Our broadcast protocol is constructed in Figure 1.

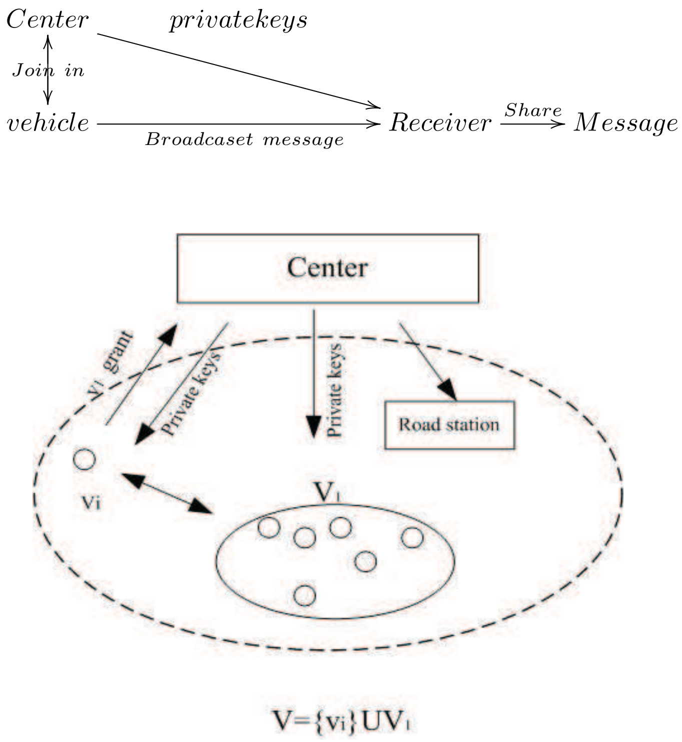


Figure 1: Broadcast Protocol in VANETs

3.1 System initialization

Suppose there are some vehicles and Infrastructure to set up a VANET. Then run it as follows. Choose α, t_1, t_2, t_3 in Z_p at random. Set $g_1 = g^{\alpha}$, $v = g^{t_1}, u = g^{t_2}, \nu = g^{t_3}$, then select $w, g_2 \in G$ at random. Compute PK, MSK as $PK = \{g, g_1, g_2, v, u, \nu, w, A = e(g_1, g_2)\}$ and $MSK = g_2^{\alpha}$.

3.2 Vehicle Grant

Let v_i denote a vehicle wants to share the data of the networks, it requests to join the VANETs V_1. Call CA and run the KeyGen to generate the private keys as follows. Let $V = v_i \bigcup V_1 = \{v_1, \cdots, v_n\}$ denote the whole vehicles in the VANETs.

Given an attribute $v_i = (v_{i1}, \cdots, v_{in}) \in V$ where $v_{ij} \in \{0,1\}$ with $j \in [1,n]$, PKG first picks $\alpha_{i1}, \cdots, \alpha_{in}$, $\beta_{i1}, \cdots, \beta_{in} \in Z_p^*$ and performs the operations in the following manner. Let $h_{i0} = g$. PKG computes auxiliary information $h_{ij} = (h_{i(j-1)})^{\alpha_i^{v_{ij}} \beta_i^{1-v_{ij}}}$, where $1 \leq j \leq n$. Let h_{kn} denote the auxiliary information of the vehicle v_k where $k \neq i$. Then compute private keys $d_{v_i} = (d_{i0}, d_{i1}, d_{i2}, d_{i3}, d_{i4}) = (g_2^{\alpha} h_{in}^{r_1}, v^{r_1}, w^{r_1}, u^{r_1 t_1}, \nu^{r_1 t_1})$, and auxiliary private keys $D_k = (h_{kn}^{r_1})$ with $1 \leq k \leq n, k \neq i$. where $r_1, r_2 \in Z_p$ and h_{in} is set as the public parameters.

3.3 Information Broadcast

To broadcast the information to infrastructure or share them in the VANETs, the vehicle encodes them and broadcasts the encrypted files to the rest of vehicles or infrastructure. At the same time, an access policy W is defined. Then perform:

Let $n = |VANETs|$ denote the scale of the VANETs. $V = (v_1, \cdots, v_n)$ denote all attributes in the VANETs where $v_i = (v_{i1}, \cdots, v_{in})$ is the attribute of the i-th vehicle. Given the access policy $W = (W_1, \cdots, W_n)$, it chooses $s_1, s_2 \in Z_p$ and computes the ciphertexts as follows.

- For $1 \leq i \leq n$, if $v_i \in W_i$, $C_0 = (\prod_{i=1}^n h_{in})^{s_1} u^{s_2}$.
- Other Ciphertexts :$C_1 = w^{s_1} \nu^{s_2}$, $C_2 = g^{s_2}, C_3 = v^{s_1}$.

The session key is set as $K = A^{s_1}$ and K will be used as encryption key to encrypt the messages by some symmetrical encryption algorithm such as AES. Finally, output the ciphertexts $C = (C_0, C_1, C_2, C_3)$

3.4 Share the Message

If a i-th vehicle in VANETs would share the files, it makes the followings. Recover

$$K = \frac{e(d_{i0} \prod_{j=1, j \neq i}^n D_j, C_3) e(d_{i3} d_{i4}, C_2) e(d_{i2}, C_3)}{e(C_1, d_{i1}) e(C_0, d_{i1})}$$

and obtain the message by using K.

The correctness can be easily checked:

$$\begin{aligned} & \frac{e(d_{i0} \prod_{j=1, j \neq i}^n D_j, C_3) e(d_{i3} d_{i4}, C_2) e(d_{i2}, C_3)}{e(C_1, d_{i1}) e(C_0, d_{i1})} \\ = & \frac{e(g_2^\alpha (\prod_{j=1}^n h_{jn})^{r_1}, v^{s_1}) e(g^{r_1(t_1 t_2 + t_1 t_3)}, g^{s_2}) e(w^{r_1}, v^{s_1})}{e(v^{r_1}, w^{s_1} \nu^{s_2}) e((\prod_{i=1}^n h_{in})^{s_1} u^{s_2}, v^{r_1})} \\ = & A^{s_1} = K. \end{aligned}$$

4 Security Reduction and Performance Analysis

4.1 Anonymous Analysis

In this section, we will prove the proposed scheme achieves not only security but also anonymity. Following [9], both of them can be achieved by distinguishing three games. The games are run as follows.

Game 1: $C = (C_0, C_1, C_2, C_3)$;

Game 2: $C = (R_0, C_1, C_2, C_3)$;

Game 3: $C = (R_0, R_1, C_2, C_3)$;

where $R_0, R_1 \in G$. Note that: for pages limits, we omit the detailed proof. If it is necessary, the detailed proof can be found in our full paper.

Lemma 1. *Game 1 and Game 2 are indistinguishable if the decisional BDHE assumption is true.*

Proof. (Sketch) We will show if there is an adversary that can distinguish between Games 1 and Game 2 with advantage ε, then the Decisional BDHE game will be solved with advantage $\frac{1}{2^n}\varepsilon$. The proof is issued from [2]. It starts from a D-BDHE challenge tuple $(g, y_0, y_1, \cdots, y_n, y_{n+2}, \cdots, y_{2n+2}, T)$ where $y_i = g^{\alpha^i}$ and T is either $T = e(g, g)^{\alpha^{n+1} c}$ or a random element of G_1. Then adversary outputs the challenge target attribute $V^* = (v_1^*, \cdots, v_n^*)$ and policies W_0^*, W_1^* . The simulator selects t_1, t_2, t_3, t, γ, $\alpha_{ij}, \beta_{ij} \in Z_p^*$ with $1 \leq i, j \leq n$, and sets up the system parameters $v = g^{t_1}, g_1 = y_1^{t_1}, u = g^{t_2}, \nu = g^{t_3}, w = g^t$ and $g_2 = y_n g^\gamma$. Output $PK = (g, g_1, g_2, u, v, w, \nu, A = e(g_1, g_2))$. The system MSK is impliedly set as $g_2^\alpha = y_1^\gamma y_{n+1}$ which keeps secretly for simulator and adversary.

For private key query to $V = (v_1, \cdots, v_n)$ with the restriction $v_i \neq v_i^*$, the followings can be simulated.

$$h_{i1} = \begin{cases} g^{\alpha_{i1}^{v_{i1}} \beta_{i1}^{1 - v_{i1}}} & \text{if } v_{i1} \neq v_{i1}^* \\ y_1^{\alpha_{i1}^{v_{i1}} \beta_{i1}^{1 - v_{i1}}} & \text{if } v_{i1} = v_{i1}^* \end{cases},$$

$$h_{i2} = \begin{cases} h_1^{\alpha_{i2}^{v_{i2}} \beta_{i2}^{1 - v_{i2}}} & \text{if } v_{i2} \neq v_{i2}^* \\ y_1^{\alpha_{i2}^{v_{i2}} \beta_{i2}^{1 - v_{i2}}} & \text{if } v_{i2} = v_{i2}^* \wedge v_{i1} \neq v_{i1}^* \\ y_2^{\alpha_{i2}^{v_{i2}} \beta_{i2}^{1 - v_{i2}}} & \text{if } v_{i2} = v_{i2}^* \wedge v_{i1} = v_{i1}^* \end{cases},$$

$$\vdots$$

where τ_i is the the number of positions such that $v_{ij} = v_{ij}^*$ in v_i ($\tau_i \neq \tau_k$ for $i \neq k$. Let $T(v_i) = \prod_{j=1}^n \alpha_{ij}^{v_{ij}} \beta_{ij}^{1 - v_{ij}}$. Then

$$h_{in} = y_{\tau_i}^{T(v_i)}.$$

The responding private keys for v_n are set as

$$d_{n0} = y_1^\gamma (y_{\tau_i}^{T(v_i)})^{r'_i}, d_{i1} = g^{r'_i t_1} y_{n - \tau_i + 1}^{-\frac{t_1}{T(v_i)}}, d_{i2} = g^{r'_i t} y_{n - \tau_i + 1}^{-\frac{t}{T(v_i)}},$$

$$d_{i3} = g^{r'_i t_1 t_2} y_{n - \tau_i + 1}^{-\frac{t_1 t_2}{T(v_i)}}, d_{i4} = g^{r'_i t_1 t_3} y_{n - \tau_i + 1}^{-\frac{t_1 t_3}{T(v_i)}},$$

where $r_i = r'_i - \frac{\alpha^{n - \tau_i + 1}}{T(v_i)}$.

The process is perfect.

$$\begin{aligned} d_{i0} &= y_1^\gamma (y_{\tau_i}^{T(v_i)})^{r'_i} = y_{n+1} y_1^\gamma (y_{\tau_i}^{T(v_i)})^{r'_i} y_{n+1}^{-1} \\ &= g_2^\alpha (y_{\tau_i}^{T(v_i)})^{r'_n} y_{n+1}^{-1} \\ &= g_2^\alpha (y_{\tau_i}^{T(v_i)})^{r'_i} (y_\tau^{T(v_i)})^{-\frac{\alpha^{n - \tau_i + 1}}{T(v_i)}} = g_2^\alpha h_{in}^{r_i}. \\ d_{i1} &= g^{r'_i t_1} y_{n - \tau_i + 1}^{-\frac{t_1}{T(v_i)}} = v^{r_i}, \\ d_{i2} &= g^{r'_i t} y_{n - \tau_i + 1}^{-\frac{t}{T(v_i)}} = w^{r_i} \\ d_{i3} &= g^{r'_i t_1 t_2} y_{n - \tau_i + 1}^{-\frac{t_1 t_2}{T(v_i)}} = u^{r_i t_1}, \\ d_{i4} &= g^{r'_i t_2 t_3} y_{n - \tau_i + 1}^{-\frac{t_2 t_3}{T(v_i)}} = \nu^{r_i t_1}. \end{aligned}$$

The auxiliary elements of private keys are set to $D_j = y_{\tau_j}^{T(v_j)r'_i} y_{n-\tau_i+\tau_j+1}^{-\frac{T(v_j)}{T(v_i)}}$

For challenge phase, according to the adversary's outputs, V^* and (W_0^*, W_1^*). the challenge ciphertext is set as $C_0 = \prod_{i=1}^n y_0^{T(v_i^*)} g^{s_2 t_2}$. where $h_{in} = g^{T(v_i^*)}$. The session key is set as $Te(y_1^\gamma, y_0)$. Otherwise, C_0 is set as random element. It also sets $(C_1^*, C_2^*, C_3^*) = (y_0^t g^{t_3 s_2}, g^{s_2}, y_0^{t_1})$. If $T = e(g,g)^{\alpha^{n+1}c}$, runs Game 1 with the adversary. Set $s_1 = c$. Then we can see that

$$\begin{aligned} K &= Te(y_1^\gamma, y_0) = e(g,g)^{\alpha^{n+1}c} e(y_1^\gamma, y_0) \\ &= e(g^{\alpha^n}, g^{\alpha c}) e(y_1^\gamma, y_0) = e(y_n, y_1)^c e(g^\gamma, y_1)^c \\ &= e(y_n g^\gamma, y_1)^c = e(g_2, g_1)^c = e(g_2, g_1)^{s_1}, \end{aligned}$$

$$C_1^* = y_0^t g^{t_3 s_2} = g^{tc} g^{t_3 s_2} = w^c \nu^{s_2} = w^{s_1} \nu^{s_2},$$

$$C_2^* = g^{s_2}, C_3^* = v^c = v^{s_1}.$$

$$C_0 = \prod_{i=1}^n y_0^{T(v_i^*)} g^{s_2 t_2} = (\prod_{i=1}^n h_{in})^{s_1} u^{s_2}.$$

Otherwise, T is a random element in G_1, and run Game 2 with the adversary.

All the process ends with the adversary outputting a guess b' .

From the [2], the simulator's advantage in the D-BDHE game is $\frac{1}{2^{n^2}}\varepsilon$. □

Lemma 2. *If the advantage of distinguishing Game 2 and Game 3 is neglected, then no t-time algorithm can solve the Decisional Linear assumption.*

Proof. This proof is similar to the Lemma 1. Note that it starts a challenge tuple

$(g, g^{z_1}, g^{z_2}, g^{z_2^2}, \cdots, g^{z_2^n}, g^{z_2^{n+2}}, \cdots, g^{z_2^{2n}}, g^{z_3}, g^{z_4}, g^{z_4 z_2}, \cdots,$ $g^{z_2^n z_4},, g^{z_1 z_2}, \cdots, g^{z_2^n z_1}, Y)$, Y is either $g^{z_1(z_3+z_4)}$ or a random element of G. □

Theorem 1. *If decisional BDHE assumption and ADL assumption hold, the proposed broadcast protocol is secure and anonymous.*

4.2 Performance Analysis

4.2.1 Efficiency Analysis

The proposed broadcast protocol has short public parameters, private keys and broadcast messages. It is a distinct feature over the available, which overcomes the weakness in the existing works since both rely on the vehicles depth. In addition, we have shown the proposed construction achieves compact security and anonymity in the standard model. Another contribution is the well dynamic behavior. VENETs are characterized by a highly dynamic topology with vehicles moving in restricted road environment with different speeds. A vehicle can move at variable speeds, pause, change lanes,turn at junctions and overtake other vehicles. Hence a vehicle may join in the networks or leave the network whenever it needs to do. In the existing works, the attribute revocation and system update are the difficult open problem especially in the broadcast system. Our protocol resolves partly the above. When a new vehicle applies to join in the networks, the center only generates the corresponding the private keys by using the communal parameters instead of resetting the system. And when a vehicle leaves the networks, the center revokes its attribute by modifying its access policy. In addition, the proposed protocol is not fixed the scale number at the beginning of the setup phase. Table 1 gives the comparison with [30].

The experiment is simulated on a modern workstation with 64-bit, 3.2 Ghz Pentium 4. The implementation uses a 224-bit elliptic curve (P224) group over a 512-bit finite field. On the test machine, the PBC library can compute bilinear pairing $e(,)$ in approximately 5.5 ms, and exponentiations in G and G_1 take about 6.4 ms and 0.6 ms respectively. It would take about 1 s to encrypt 6 M-bit messages by using AES. Hence considering the 1000-bit message which is used to send, the operation time by AES in receiving devices is negligible. Let $n = 20$ and some comparisons are given in Table 2.

4.2.2 Better Performance-An Extension

For achieving anonymity, access policy must be hidden in the proposal, which results in performing the whole decryption procedure in order to verify whether he/she is the intended receiver of the ciphertext or not [17][36]. In order to overcome this weakness, a matching verification before decryption is performed. A new scheme is given as follows.

System initialization. Suppose there are some vehicles and Infrastructure to set up a VANET. The parameters are generated as basic one.

Vehicle Grant. As basic one.

Message Broadcast. Let $n = |VANETs|$ denote the scale of the VANETs. $V = (v_1, \cdots, v_n)$ denote all attributes in the VANETs where $v_i = (v_{i1}, \cdots, v_{in})$ is the i-th user. Given the access policy $W = (W_1, \cdots, W_n)$, it chooses $s_1, s_2 \in Z_p$ and computes the ciphertexts as follows.

- Matching ciphertexts with W: If $v_i \in W_i$, $C_{0i} = h_{in}^{s_1} u^{s_{2i}}, C_{2i} = g^{s_{2i}}$, where $s_{2i} \in Z_p$ and $\sum_{i=1}^k s_{2i} = s_2$. Otherwise C_{0i}, C_{2i} is set as a random element in G.
- Other Ciphertexts: $C_1 = w^{s_1} \nu^{s_2}$, $C_3 = v^{s_1}$.

The session key is set as $K = A^{s_1}$ and K will be used as encryption key to encrypt the messages by some symmetrical encryption algorithm such as AES. Finally, output the ciphertexts $C = (C_{0i}, C_1, C_{2i}, C_3)_{1 \leq i \leq n}$

Table 1: Comparisons with [30]

Scheme	Broadcast (Sending)	Share (Receiving)	Hash	Sign	Revocation	Dynamic feature
[30]	O(1)	O(1)	Yes	Yes	No	No
Ours	O(1)	O(1)	NO	NO	Yes	Yes

Table 2: Comparisons of computational efficiency with [30]

	[5]	Ours
Broadcast (Sending)	31.05 ms	38.4 ms
Share(Receiving)	62.16 ms	9.25 ms

Share the Message. Besides basic one proceeds, the sharing user performs as follows.

Let $T(v_i) = \prod_{j=1}^{n} \alpha_{ij}^{v_{ij}} \beta_{ij}^{1-v_{ij}}$. the matching operations is performed as

$$1 = \frac{e(d_{i4}, \prod_{i=1}^{n} C_{2i}) e(d_{i2}, C_3)}{e(C_1, d_{i1})}.$$

If it holds, compute

$$K = \frac{e(d_{i0} \prod_{j=1, j\neq i}^{n} D_j, C_3) e(d_{i3} d_{i4}, \prod_{i=1}^{n} C_{2i}) e(d_{i2}, C_3)}{e(C_1, d_{i1}) e(\prod_{i=1}^{n} C_{0i}, d_{i1})}.$$

Then it recovers the message by using K.

5 Conclusions

In this paper, we propose a new broadcast protocol. Compared with previous scheme, it achieves a good trade-off between ciphertexts and privacy protection. The proposed scheme has efficient Broadcasting algorithm and Shareing message algorithm in terms of computation cost. In addition, it supports the dynamic management of the VANETs and does not fixed the scale of the whole networks. The constant size of ciphertexts decreases the communication cost and storage cost which is important for the resource-constrained VANETs.

Acknowledgments

This work was supported in part by the National Nature Science Foundation of China under Grant (NO. U19B2021), Key Research and Development Program of Shaanxi (No 2022GY-028, 2022GY-050).

References

[1] M. Abdalla, M. Bellare, D. Catalano, "Searchable encryption revisited: consistency properties, relation to anonymous IBE, and extensions," in *Proceedings of Advances in Cryptology (CRYPTO'05)*, LNCS, vol. 3621, Springer-Verlag, pp. 205-222, 2005.

[2] M. Abdalla, D. Catalano, and D. Fiore, "Verifiable Random Functions from Identity-based Key Encapsulation," in *Proceedings of Advances in Cryptology (Eurocrypt'09)*, LNCS, vol. 5479, pp. 554-571, 2009.

[3] W Ahmed, M, Elhadef, "Securing Intelligent Vehicular Ad Hoc Networks: A Survey," *Advances in Computer Science and Ubiquitous Computing, Lecture Notes in Electrical Engineering*, vol. 474, pp. 6-14, 2018.

[4] A. Benslimane, "Optimized dissemination of alarm messages in vehicular ad-hoc networks (VANET)," in *Proceedings of the 7th IEEE International Conference High Speed Networks and Multimedia Communications*, pp. 655-666, 2004.

[5] A. R. Beresford, F. Stajano, "Location Privacy in Pervasive Computing," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 46-55, 2003.

[6] J. Bethencourt, A. Sahai, B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of Symposium on Security and Privacy*, pp. 321-334, IEEE press, 2007.

[7] D. Boneh, X. Boyen, and H. Shacham, "Short group signature," in *Proceedings of Advances in Cryptology*, LNCS, vol. 3152, Springer-Verlag, pp. 41-55, 2004.

[8] D. Boneh, M. Franklin, "Identity-based encryption from the weil pairing," in *Proceedings of Advance in Cryptology (CRYPTO'01)*, LNCS, vol. 2139, Springer-Verlag, pp. 213-229, 2001.

[9] X. Boyen, B. Waters, "Anonymous hierarchical identity-based encryption (without random oracles)," in *Proceedings of Advances in Cryptology (CRYPTO'06)*, LNCS, vol. 5677, pp. 290-317, 2006.

[10] C. Büttner, S. A. Huss, "An Efficient Anonymous Authenticated Key Agreement Protocol for Vehicular Ad-Hoc Networks Based on Ring Signatures and the Elliptic Curve Integrated Encryption Scheme," *ICISSP*, pp. 139-159, 2015.

[11] E. F. Cahyadi, C. Damarjati, M. S. Hwang, "Research on identity-based batch verification schemes for security and privacy in VANETs", *Journal of Electronic Science and Technology*, vol. 20, no. 3, pp. 1-19, 2022.

[12] E. F. Cahyadi, M. S. Hwang, "A comprehensive survey on certificateless aggregate signature in vehicular ad hoc networks", *IETE Technical Review*, vol. 39, no. 6, pp. 1265-1276, 2022.

[13] E. F. Cahyadi, M. S. Hwang, "An improved efficient anonymous authentication with conditional privacy-preserving scheme for VANETs", *Plos One*, vol. 16, no. 9, 2021.

[14] E. F. Cahyadi, M. S. Hwang, "An improved efficient authentication scheme for vehicular ad hoc networks with batch verification using bilinear pairings", *International Journal of Embedded Systems*, vol. 15, no. 2, pp. 139-148, 2022.

[15] E. F. Cahyadi, M. S. Hwang, "A lightweight BT-based authentication scheme for illegal signatures identification in VANETs", *IEEE Access*, vol. 10, pp. 133869-133882, 2022.

[16] E. F. Cahyadi, T. W. Su, C. C. Yang, M. S. Hwang, "A certificateless aggregate signature scheme for security and privacy protection in VANET", *International Journal of Distributed Sensor Networks*, vol. 18, no. 5, 2022.

[17] P. Chaudhari, M. L. Das, and A. Mathuria, "On Anonymous Attribute Based Encryption," *ICISS, LNCS*, vol. 9478, Springer-Verlag, pp. 378-392, 2015.

[18] B. K. Chaurasia, S. Verma, S. M. Bhasker, "Message broadcast in VANETs using Group Signature," in *International Conference on Wireless Communication and Sensor Networks*, pp. 131-136, 2009.

[19] P. S. Chung, C. W. Liu, M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments", *International Journal of Network Security*, vol. 16, no. 1, pp. 1-13, 2014.

[20] M. Durresi, A. Durresi, L. Barolli, "Emergency broadcast protocol for inter-vehicle communications," in *Proceedings of the 11th International Conference on Parallel and Distributed Systems*, pp. 402-406, 2005.

[21] E. C. Eze, S. Zhang, E. Liu, and Joy C. Eze, "Advances in Vehicular Ad-hoc Networks (VANETs): Challenges and Road-map for Future Development," *International Journal of Automation and Computing*, vol. 13, no. 1, pp. 1-18, 2016.

[22] E. Fasolo, R. Furiato, A. Zanella, "Smart Broadcast algorithm for inter-vehicular communications," *Proceedings of the Wireless Personal Multimedia Communication*, 2005.

[23] B. Gedik, L. Liu, "Location Privacy in Mobile Systems: A Personalized Anonymization Model," in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems*, 2005.

[24] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," *ACM CCS*, ACM press, pp. 89-98, 2006.

[25] M. Gruteser, D. Grunwald, "Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking," in *Proceedings of the First International Conference on Mobile Systems, Applications and Services*, 2003.

[26] G. Hu, L. Zhang, Y. Mu and X. Gao, "An Expressive Test-Decrypt-Verify Attribute-Based Encryption Scheme With Hidden Policy for Smart Medical Cloud," *IEEE Systems Journal*, vol.15, pp. 365-376, 2021.

[27] H. Jiang, H. Guo, L. J. Chen, "Reliable and efficient alarm message routing in VANET," in *Proceedings of the 28th International Conference on Distributed Computing Systems Workshops*, pp. 186-191, 2008.

[28] T. H. Kim, W. K. Hong, H. C. Kim, "An effective multihop broadcast in vehicular ad-hoc network," in *Proceedings of the 20th International Conference*, pp. 112-125, 2007.

[29] G. Korkmaz, E. Ekici, F. Özgüner, "Urban multihop broadcast protocol for inter-vehicle communication systems," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*, pp. 76-85, 2004.

[30] C. Laurendeau, M. Barbeau, "Secure Anonymous Broadcasting in Vehicular Networks," in *32-nd IEEE Conference on Local Computer Networks*, pp. 661-668, 2007.

[31] C. T. Li, M. S. Hwang, Y. P. Chu, "A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks", *Computer Communications*, vol. 31, no. 12, pp. 2803-2814, 2008.

[32] C. W. Liu, W. F. Hsien, C. C. Yang, M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage", *International Journal of Network Security*, vol. 18, no. 5, pp. 900-916, 2016.

[33] R. Ma, L. Zhang, Q. Wu, F. Rezaeibagha, "BE-TRDSS: Blockchain-Enabled Secure and Efficient Traceable-Revocable Data-Sharing Scheme in Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, 2023. (doi: 10.1109/TII.2023.3241618)

[34] J. H. Park, D. H. Lee, "Anonymous HIBE: Compact Construction Over Prime-Order Groups," *IEEE Transactions on Information Theory*, vol. 59, no. 4, pp. 2531-2541, 2013.

[35] C. E. Plazzi, S. Ferretti, M. Roccetti, G. Pau, M. Gerla, "How do you quickly choreograph inter-vehicular communications? A fast vehicleto-vehicle multi-hop broadcast algorithm, explained," in *Proceedings of the 4th IEEE Consumer Communications and Networking Conference*, pp. 960-964, 2007.

[36] M. K. Reiter, A. D. Rubin, "Anonymous Web Transactions With Crowds," *Communications of the ACM*, vol. 42, no. 2, pp. 32-48, 1999.

[37] A Sahai, and B Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology (EUROCRYPT'05)*, pp. 457-473, Springer-Verlag, 2005.

[38] A. Singh, H. C. S. Fhom, "Restricted usage of anonymous credentials in vehicular ad hoc networks for misbehavior detection," *Int. J. Inf. Secur.*, Sept. 2016. (DOI 10.1007/s10207-016-0328-y)

[39] J. Su, L. Zhang, Y. Mu, "BA-RMKABSE: Blockchain-aided Ranked Multi-keyword Attribute-based Searchable Encryption with Hiding Policy for Smart Health System," *Future Generation Computer Systems*, vol. 132, pp. 299-309, 2022.

[40] M. T. Sun, W. C. Feng, T. H. Lai, K. Yamada, H. Okada, K. Fujimura, "GPS-based message broadcast for adaptive intervehicle communications," in *Proceedings of the 52nd Vehicular Technology Conference*, vol. 6, pp. 2685-2692, 2000.

[41] P. Syverson, D. Goldschlag, and M. Reed, "Anonymous Connections and Onion Routing," in *Proceedings of the IEEE Symposium on Security and Privacy*, 1997.

[42] M. Torrent-Moreno, "Inter-vehicle communications: Assessing information dissemination under safety constraints," in *Proceedings of the 4th Annual Conference on Wireless on Demand Network Systems and Services*, IEEE, pp. 59-64, 2007.

[43] W. Yang, "Security in Vehicular Ad Hoc Networks," in *Wireless Network Security*, pp. 95-128, 2013.

[44] N. Zhang, Q. Shi, M. Merabti, "Anonymous Public-Key Certificates for Anonymous and Fair Document Exchange," *IEE Proceedings Communications*, vol. 147, no. 6, pp. 345-350, 2000.

[45] Y. Zhang, X. Chen, J. Li, D. S. Wong and H. Li, "Anonymous attribute-based encryption supporting efficient decryption test," in *Proceedings of the ACM SIGSAC Symposium on Information, Computer and Communications Security*, pp. 511-516, 2013.

Biography

Guang, Yang is a PH.D candidate in the school of mathematics and statistics at Xidian University, Xi'an China. His current research interests include network security, and cryptography

Leyou Zhang is a professor in the school of mathematics and statistics at Xidian University, Xi'an China. He received his PhD from Xidian University in 2009. From Dec. 2013 to Dec. 2014, he is a research fellow in the school of computer science and software engineering at the University of Wollongong. His current research interests include network security, computer security, and cryptography.

Ruonan Ma is a postgraduate in the school of mathematics and statistics at Xidian University, Xi'an China. Her current research interests include network security, and cryptography

.

A Study on Privacy Protection under Blockchain Data Transaction Based on Legal Perspective

Yajie Zhang

(Corresponding author: Yajie Zhang)

Law Teaching and Research Department, Party School of the CPC Shijiazhuang Municipal Committee
Shijiazhuang, Hebei 050000, China
jiepian84888@yeah.net

(Received Apr. 8, 2022; Revised and Accepted Mar. 23, 2023; First Online Apr. 27, 2023)

Abstract

The privacy protection issue in blockchain data transactions has received much attention. This paper first briefly analyzed the privacy protection issue in data transactions and then explained the current legal problems from the legal perspective. After introducing blockchain technology, the homomorphic encryption method was applied to protect privacy under blockchain data transactions. A privacy protection method based on the Paillier algorithm was designed, and the method's performance was analyzed. After verifying the correctness of the algorithm, the comparison with the zkSNARKs technique revealed that our method had better efficiency. When the key was 2,048 bits, the encryption homomorphism time was 0.17 ms, and the encryption and decryption time was 185.75 ms and 256.12 ms, i.e., the method had a slight burden on the system and could meet the needs of practical use. The results verify that the designed method was reliable. Therefore, it can be applied in actual blockchain data transactions.

Keywords: Blockchain; Data Transaction; Law; Paillier Algorithm; Privacy Protection

1 Introduction

As society continues to develop toward informationization, the importance of data is becoming more and more prominent. In people's production and life, a large amount of data is generated, and data transaction, which is gradually becoming a method of data circulation, provides services for scientific research, social management, etc. [9]. However, there is a lot of information related to personal privacy in the data, and it is easy to be illegally leaked in the process of transaction; therefore, privacy protection under data transactions has become a matter of great concern. From the legal point of view, illegal trading, illegal dumping, and illegal leakage in the process of data transaction involve the infringement of personal information and privacy, etc. Although data is not a tangible object, data transaction has much in common with the act of buying and selling and can be analyzed by referring to the legal provisions of the act of buying and selling.

The two subjects in the data transaction need to transmit the real and valid data securely and use it legally, thus ensuring a healthy data flow. However, there are still many loopholes in the legal regulation of data transactions, and the relevant legal provisions are not clear enough, which makes it very difficult to protect privacy under data transactions. In this case, in order to improve the security of data transactions, we can start from the technical aspect. Blockchain, as a decentralized, open and transparent technology [4, 21], has good security, but blockchain is not designed for data transactions [3,7]. Further research and design are still needed for privacy protection under blockchain data transactions [6, 22]. Onik et al. [17] proposed a blockchain-based personal identifiable information management system to limit the leakage problem of personal data and verified the privacy of the approach. Wang et al. [25] proposed a dual blockchain privacy protection approach to achieve secure data interaction between users, doctors, and hospitals through user chains and medical chains and found through experiments that the approach had low communication overhead.

Devidas et al. [8] designed a decentralized group signature scheme for solving the trust problem of centralized group managers and the privacy problem of users and verified the security and correctness of the scheme. Nóbrega et al. [16] proposed a privacy preserving record linking method based on blockchain technology and verified the effectiveness of the method with real-world data sources. This paper mainly studied the privacy protection problem under blockchain data transactions, designed a privacy protection method based on the Paillier algorithm, and verified the reliability of the method through experiments. This work provides a new method to further improve the security of blockchain and ensure the privacy of data transactions.

2 Blockchain Privacy Protection Method Based on the Paillier Algorithm

2.1 Data Transactions and the Law

With the development of the network, data is also becoming more and more a new productivity. The research and analysis of data can obtain many valuable information, so it has been applied in recommendation systems, user analysis, etc. [14], but in this process, it is likely to cause infringement on privacy [12]. In the process of data transaction, it is more vulnerable to illegal attacks and tampering, leading to the leakage of private information.

According to the Personal Information Protection Law, there are personal data right and data property right, i.e., the data subject enjoys the right to decide whether his or her data can be collected and used and also has the right to request the data transaction process to ensure the privacy of personal data, inquire about the data, and delete the data, etc. However, there are still many problems in the current law: personal information data is not clearly defined, and defending the rights is difficult. Therefore, in this case, it is especially important to realize the privacy protection under blockchain data transactions through technical means.

2.2 Blockchain Technology

Blockchain technology combines distributed networks, digital signatures and other technologies [13,19,20], which is used for recording all transactions and time of all blockchain nodes. In the public ledger, every transaction is verified by consensus of most participants, and the transaction information will be permanently written in the blockchain and will not be deleted [24]. The structure of the blockchain is shown in Figure 1.

Nodes generate the Merkle root through the Hash algorithm and store it in the block head. All transactions in the blockchain are permanently stored in the blockchain through digital signatures. Data transactions under the blockchain include five steps:

1) Generate transactions: user nodes digitally sign through private keys, generate transactions, and wait for miner nodes to pack;
2) Miner nodes randomly select transactions, verify digital signatures, and pack legitimate transactions into chunks;
3) Calculate the hash value of the block, create locally packed blocks as new blocks and broadcast them to the whole network;
4) Verify new blocks and save legal new blocks into the local block chain;
5) Write the transactions into the block chain permanently to avoid malicious tampering.

The consensus mechanism is the core of the blockchain, which is used to ensure the consistency of the blockchain. Common consensus mechanisms are as follows.

1) Proof of Work (PoW) mechanism [10]: Based on the workload, the nodes are rewarded, which has high credibility.
2) Proof-of-interest (PoS) mechanism [2]: It can effectively avoid malicious attacks.
3) Delegated proof-of-stake (DPoS) mechanism [23]: It reduces the size of committers, which is efficient.
4) Practical Byzantine fault-tolerant (PBFT) consensus algorithm [15]: It is computation-based and has high consensus efficiency.

Compared with the centralized system, the decentralized feature of blockchain makes it more advantageous in terms of privacy protection, but it does not guarantee absolute privacy. Due to the transparency of transaction information on the chain, there is a threat to the user's identity privacy and transaction privacy; therefore, privacy protection under blockchain data transactions is also a very important issue.

2.3 Homomorphic Encryption and the Paillier Algorithm

Homomorphic encryption can ensure the correctness of operations while securing the original data [11], which has a wide range of applications in privacy protection. It is assumed that there is plaintext space M, ciphertext space C, and key space K, and the encryption and decryption algorithms are E and D. If there is $P(E_k(m_1), E_k(m_2), \cdots, E_k(m_n)) = E_k(L(m_1, m_2, \cdots, m_n))$, then the encryption scheme is homomorphic. For different operations L, there are several scenarios.

For any plaintext m_i, $m_j \in M$, the corresponding ciphertext $c_i = E(m_i)$, $c_j = E(m_j)$, and $c_(i), c_j \in C$, if

1) $E(m_i + m_j) = E(m_i) \oplus E(m_j)$ or $D(E(m_i) \oplus E(m_j)) = m_i + m_j$, it is an addition homomorphism;
2) $E(m_i m_j) = E(m_i) \oplus E(m_j)$ or $D(E(m_i) \oplus E(m_j)) = m_i m_j$, it is a multiplication homomorphism;
3) $E(m_i m_j) = E(m_i) \oplus m_j$ or $D(E(m_i) \oplus m_j) = m_i m_j$, it is a mixed multiplication homomorphism.

The Paillier algorithm is an algorithm that satisfies addition homomorphism [1], and its definition is as follows. Big prime numbers p and q are randomly chosen to make $\gcd(pq, (p-1)(q-1)) = 1$, and gcd is the greatest common divisor. Then, $n = pq$ and $\lambda = lcm(p-1, q-1)$ are calculated, where lcm is the least common multiple. An integer

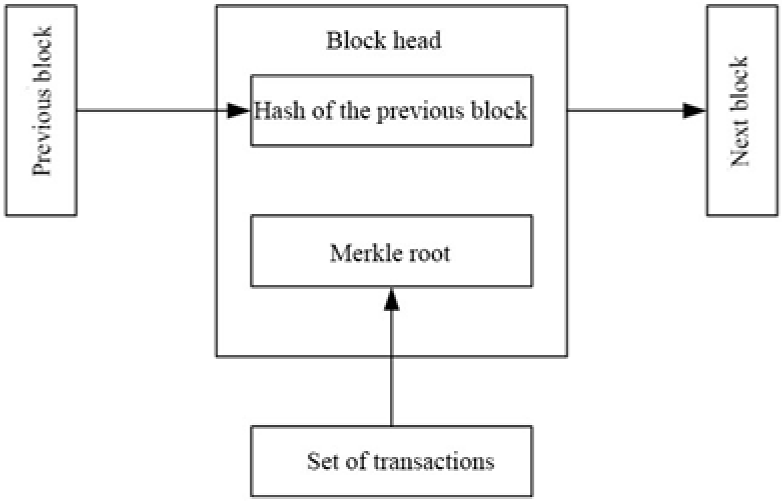


Figure 1: Blockchain structure

is randomly selected: $g \in Z_(n^2)^*$, $\mu = (L(g^\lambda \bmod n^2))^{-1}$, $L(x) = (x-1)/x$, then, public key $PK = (n, g)$ and private key $SK = (\lambda, \mu)$ are obtained. For the message to be encrypted $m(0 \leq m < n)$, integer $r \in Z_(n^2)^*$ is randomly selected to ensure $\gcd(r, n) = 1$. Then, the encryption process is written as: $c \leftarrow E(m, PK)$, $E(m, PK) = g^m r^n \bmod n^2$, and the decryption process is written as: $m \leftarrow D(c, SK)$, $D(c, SK) = L(c^\lambda \bmod n^2)\mu \bmod n^2$.

According to the addition homomorphism property of the Paillier algorithm, if $c_1 \leftarrow E(m_1, PK)$, $c_2 \leftarrow E(m_2, PK)$, then:

$$\begin{aligned} c_1 \times c_2 &= g^(m_1)r_1^n \times g^(m_2)r_2^n \bmod n^2 \\ &= g^(m_1 + m_2)(r_1 r_2)^n \bmod n^2 \\ &= E(m_1 + m_2, PK). \end{aligned}$$

2.4 Privacy Protection Method Based on the Paillier Algorithm

The Paillier algorithm is applied to privacy protection under blockchain data transactions [5]. Suppose that in the blockchain, users A and B conduct transactions, the initial amounts deposited by A and B are B_0 and B_1, respectively, and the total account balance is B, the transfer amount of every transaction is $\{v_1, v_2, \cdots, v_n\}$ in the process of conducting multiple transactions $\{T_1, T_2, \cdots, T_n\}$, and the remaining amounts of A and B are $b_(0, n)$ and $b_(1, n)$ at the end of the transaction. Based on the Paillier algorithm, big prime numbers a and b are randomly selected to ensure $\gcd(ab, (a-1)(b-1)) = 1$, $n = ab$ and $\lambda = lcm(a-1, b-1)$ are calculated, and then integer $g \in Z_(n^2)^*$ is randomly chosen. $\mu = (L(g^\lambda \bmod n^2))^{-1}$ is calculated, and public key $PK = (n, g)$ and private key $SK = \lambda$ are obtained. The public-private key pair of A is (PK_0, SK_0), and the public-private key pair of B is (PK_1, SK_1).

Both A and B encrypt the balance deposited into the joint account to obtain M_1 and M_2. $E(M_1, M_2)$ is decrypted by Certificate Authority (CA), and the decryption process is $E(M_1, M_2) = (g^(B_0)r_1^n \bmod n^2)(g^(B_1)r_2^n \bmod n^2) = [g^(M_1 + M_2)(r_1 r_2)^n \bmod n^2]$.

In the first transaction, A sends the amount of the first transfer (v_1) and the account balance $(b_0, 1)$ to CA to verify T_1 after encryption. The ciphertexts after encryption are: $M_1 = g^(v_1)r_3^n \bmod n^2$, $M_2 = g^(b_0, 1)r_4^n \bmod n^2$. According to the property of addition homomorphism of the Paillier algorithm, if $m_1 \times m_2 = M_1$, then $b_0, 1 + v_1 = B_0$.

At the end of the transaction, A and B encrypt the current account balance and send it to CA for verification, and the encrypted ciphertexts are $M_3 = g^(b_(0, n))r_5^n \bmod n^2$, $M_4 = g^(b_(1, n))r_6^n \bmod n^2$. If $E(M_3, M_4) = B$, then the CA closes the payment channel after writing the encrypted balance to the blockchain, i.e., the transaction is completed.

3 Experiment and Analysis

From the perspective of the correctness of the algorithm, taking the first transaction as an example, CA determines whether the transaction is legitimate or not by verifying the correctness of $m_1 \times m_2 = M_1$. The proof of the process is as follows:

$$\begin{aligned} m_1 &= E(v_1, r_3) \\ &= g^(v_1)r_3^n \bmod n^2 \\ m_2 &= E(b_0, 1, r_4) \\ &= g^(b_0, 1)r_4^n \bmod n^2 \\ m_1 \times m_2 &= g^(v_1)r_3^n \bmod n^2 g^(b_0, 1)r_4^n \bmod n^2 \\ &= g^(v_1 + b_0, 1)(r_3 r_4)^n \bmod n^2 \\ &= E(v_1 + b_0, 1, r). \end{aligned}$$

Through the above calculation, b_0, $1+v_1 = B_0$ is verified; therefore, the Paillier algorithm is theoretically correct.

In order to understand the effectiveness of the privacy protection method based on the Paillier algorithm, experiments were conducted in Windows 10 environment and 8 GB memory. At the same time, the method was compared with the Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zkSNARKs) technique based on the zero-knowledge proof library libsnark [18]. The core of the zkSNARKs technique is the elliptic encryption algorithm, whose cryptosystem is different from the Paillier algorithm. Therefore, in the comparison process, the privacy protection methods under two blockchain data transactions were compared by controlling the number of keys of the Paillier algorithm and the number of constraints of libsnark.

First, the number of keys of the Paillier algorithm was set as 64 bits, 128 bits, 256 bits, 512 bits, 1,024 bits and 2,048 bits, respectively. Depending on the key sizes, the time required for the addition homomorphism of the Paillier algorithm, encryption, and decryption are shown in Table 1.

It was seen from Table 1 that the addition homomorphic time grew slowly with the increase of key size, but the change was small. The additive homomorphic time of the Paillier algorithm was 0.07 ms when the key was 64 bits and 0.17 ms when the key was 2,048 bits, indicating an increase of 0.1 ms. This suggested that the addition homomorphic time was little affected by the key, which meant that CA could complete the verification of the transaction with a low burden during the blockchain data transaction.

Then, the encryption and decryption time of the Paillier algorithm increased significantly with the increase of key size. When the key was 64 bits, the encryption time of the Paillier algorithm was 0.56 ms; when the key increased to 2,048 bits, the encryption time increased to 185.75 ms, which was about 331 times of that when the key was 64 bits. In addition, it was found that the encryption time increased mildly when the key increased from 64 bits to 512 bits, and when it increased from 1,024 bits to 2,048 bits, the encryption time showed a significant increase. Similarly, the decryption time of the algorithm increased from 0.59 ms at 64 bits to 256.12 ms at 2048 bits, which showed an increase of about 434 times. This indicated that the key change had a great impact on the encryption and decryption time, and a significant increase occurred after 1,024 bits. The decryption time was slightly longer than the encryption time. Taking 2048 bits for example, the decryption time was 256.12 ms, which was 70.37 ms longer than the encryption time. In general, the addition homomorphic time ¡ encryption time ¡ decryption time in the case of the same key.

Then, the efficiency of the zkSNARKs technique was analyzed. Different number of libsnark constraints were set, and the generation time and verification time of zero-knowledge proofs are shown in Table 2.

It was seen from Table 2 that the zkSNARKs technique took much time to generate zero-knowledge proofs, and the generation time reached 0.21 s when the number of constraints was only 1,000 and 9.87 s when the number of constraints was 50,000. Although the time required by the zkSNARKs technique for proof verification was only milliseconds, overall, its efficiency was significantly higher than that of the Paillier algorithm, which imposed a large burden on the system; therefore, the privacy protection method based on the Paillier algorithm was more applicable.

4 Discussion

With the construction of digitalization and informatization, the demand for data transactions has been growing, while at the same time, the issue of privacy protection under data transactions has gradually surfaced, reflecting the inadequacy of the current legal regulation. From the legal point of view, the provider, the buyer and the intermediary party of data transactions has a legal relationship, which is closely related to personal privacy. In June 2018, a user on the dark web sold 1 billion pieces of express data from YTO Express, which involves personal information such as the name and phone number of the user. Moreover, the behavior of collecting personal information exists in many APPs. These issues seriously threaten personal privacy; therefore, the legal regulation of data transaction is very important. For data transactions, the relevant laws are as follows.

1) According to the Network Security Law, data are generally traded with the network as the medium, then the network service provider must strictly comply with relevant laws, strengthen industry self-regulation, and keep honest and faith.

2) The Data Security Law of the People's Republic of China regulates the processing and exploitation of data, providing a new legal basis for the regulation of data transactions.

3) The Personal Information Protection Act regulates the trading of personal data to a certain extent.

According to the current legal regulation, the shortcomings are as follows.

1) The ownership of data property is still unclear. In the process of data trading, many information that has been anonymized is considered not to be private and cannot be protected according to the personal information protection law; moreover, as a kind of content that can be copied and disseminated infinitely, the subject of data rights is also difficult to be clarified.

2) There is no special legislation on data trading yet, and only some guidelines exist in other related laws, which have strong limitations.

3) There is no special supervisory department to supervise data trading.

Table 1: Efficiency analysis of the Paillier algorithm

Key size	Addition homomorphic time/ms	Encryption time/ms	Decryption time/ms
64 bits	0.07	0.56	0.59
128 bits	0.08	0.79	0.92
256 bits	0.09	2.46	2.56
512 bits	0.11	5.69	6.72
1,024 bits	0.13	27.65	35.41
2,048 bits	0.17	185.75	256.12

Table 2: Efficiency analysis of the zkSNARKs technique

Number of constraints	Zero knowledge proof generation time/s	Zero knowledge proof verification time/ms
1000	0.21	1.21
5000	1.25	5.64
10000	2.07	10.3
3 20000	3.56	11.8
9 30,000	4.69	13.5
6 40000	7.55	14.17
50,000	9.87	17.21

4) It relies too much on platform self-regulation, and the relevant regulation is not yet sound.

From the technical perspective, this paper studied the privacy protection problem under blockchain data transactions, designed a privacy protection method based on the Paillier algorithm, and verified the reliability of the method through experimental analysis, which makes some contributions to further realize the security of data transactions. In the face of the continuous development of data transactions, in addition to the technical level, the legal level should also be taken account to strengthen the supervision; therefore, this paper puts forward the following suggestion:

1) Further clarify the ownership of data property and protect the rights and interests of data owners, generators, users, practitioners, etc.;

2) Further improve the regulation of data transactions, improve the importance and recognition of data, and establish relevant regulatory sections to form a well-organized regulatory system;

3) Strengthen the legal regulation of data transactions at the level of civil, administrative, and criminal law to ensure the legitimate rights and interests of individuals while maximizing the use of data.

5 Conclusion

This paper mainly studied the privacy protection under blockchain data transactions, elaborated on the relevant legal knowledge, and then designed a method based on the Paillier algorithm to enhance the security of data transactions under blockchain. It was found through experimental analysis that the method had good correctness and also high computational efficiency, so compared with the zkSNARKs technique, it was less burdensome to the system. The proposed method can be promoted and applied in the actual blockchain to promote the security of data transactions.

References

[1] M. Alanezi, "Enhancing cloud computing security by paillier homomorphic encryption," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 2, pp. 1771, 2021.

[2] K. Bala, P. D. Kaur, "A novel game theory based reliable proof-of-stake consensus mechanism for blockchain," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 9, pp. 1-24, 2022.

[3] P. Y. Chang, M.S. Hwang, C.C. Yang, "A blockchain-based traceable certification system", in *Security with Intelligent Computing and Big-data Services*, pp. 363-369, 2018.

[4] L. Chen, Q. Fu, Y. Mu, L. Zeng, F. Rezaeibagha, M. S. Hwang, "Blockchain-based random auditor committee for integrity verification", *Future Generation Computer Systems*, vol. 131, pp. 183-193, 2022.

[5] Y. C. Chen, W. L. Wang, M. S. Hwang, "RFID authentication protocol for anti-counterfeiting and pri-

vacy protection", in *The 9th International Conference on Advanced Communication Technology*, pp. 255-259, 2007.

[6] Y. H. Chen, L. C. Huang, I. C. Lin, M. S. Hwang, "Research on the secure financial surveillance blockchain systems", *International Journal of Network Security*, vol. 22, no. 4, pp. 708-716, 2020.

[7] Y. H. Chen, L. C. Huang, I. C. Lin, M. S. Hwang, "Research on blockchain technologies in bidding systems", *International Journal of Network Security*, vol. 22, no. 6, pp. 897-904, 2020.

[8] S. Devidas, S. Rao, N. R. Rekha, "A decentralized group signature scheme for privacy protection in a blockchain," *International Journal of Applied Mathematics and Computer Science*, vol. 31, no. 2, pp. 353-364, 2021.

[9] A. Dunn, K. Hood, A. Batch, A. Driessen, "Measuring consumer spending using card transaction data: lessons from the COVID-19 pandemic," *AEA Papers and Proceedings*, vol. 111, pp. 321-325, 2021.

[10] X. Feng, J. Ma, S. Liu, Y. Miao, X. Liu, "Auto-scalable and fault-tolerant load balancing mechanism for cloud computing based on the proof-of-work election," *Science China Information Sciences*, vol. 65, no. 1, pp. 112102, 2021.

[11] C. Y. Lai, K. M. Chung, "On statistically-secure quantum homomorphic encryption," *Quantum Information & Computation*, vol. 18, no. 9 &10, pp. 785-794, 2018.

[12] F. Loukil, C. Ghedira-Guegan, K. Boukadi, A. N. Benharkat, "Towards an End-to-End IoT data privacy-preserving framework using blockchain technology," in *International Conference on Web Information Systems Engineering*, pp. 68-78, 2018.

[13] C. Y. Lin, L. C. Huang, Y. H. Chen, M. S. Hwang, "Research on security and performance of blockchain with innovation architecture technology", *International Journal of Network Security*, vol. 23, no. 1, pp. 1-8, 2021.

[14] M. A. Morris, E. L. Wilkins, M. Galazoula, S. Clark, M. Birkin, "Assessing diet in a university student population: A longitudinal food card transaction data approach," *British Journal of Nutrition*, vol. 123, no. 12, pp. 1406-1414, 2020.

[15] G. I. Navaroj, E. G. Julie, Y. H. Robinson, "Adaptive practical Byzantine fault tolerance consensus algorithm in permission blockchain network," *International Journal of Web and Grid Services*, vol. 18, no. 1, pp. 62-82, 2022.

[16] T. Nóbrega, C. Pires, D. C. Nascimento, "Blockchain-based privacy-preserving record linkage enhancing data privacy in an untrusted environment," *Information Systems*, vol. 102, no. 6, pp. 1-19, 2021.

[17] M. Onik, C. S. Kim, N. Y. Lee, J. Yang, "Privacy-aware blockchain for personal data sharing and tracking," *Open Computer Science*, vol. 9, no. 1, pp. 80-91, 2019.

[18] J. Park, H. Kim, G. Kim, J. Ryou, "Smart contract data feed framework for privacy preserving orac1e system on b1ockchain," *Computers*, vol. 10, no. 1, pp. 1-12, 2020.

[19] Z. P. Peng, M. L. Chiang, I. C. Lin, C. C. Yang, M. S. Hwang, "CBP2P: Cooperative electronic bank payment systems based on blockchain technology", *Journal of Internet Technology*, vol. 23, no. 4, pp. 683-692, 2022.

[20] S. Seybou, F. Essaf, M. Mbyamm, "Privacy protection issues in blockchain technology," *International Journal of Computer Science and Information Security*, vol. 17, no. 2, pp. 124-131, 2019.

[21] E. Sharma, "A framework of big data as service platform for access control & privacy protection using blockchain network," *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 11, pp. 476-485, 2021.

[22] M. Storm, L. Moerel, "Blockchain can both enhance and undermine compliance but is not inherently at odds with EU privacy laws," *Journal of Investment Compliance*, vol. 22, no. 2, pp. 122-132, 2021.

[23] Y. Sun, B. Yan, Y. Yao, J. Yu, "DT-DPoS: A delegated proof of stake consensus algorithm with dynamic trust," *Procedia Computer Science*, vol. 187, no. 9, pp. 371-376, 2021.

[24] S. Tsakiridi, "Blockchain and the GDPR - friends or foes?," *Privacy & Data Protection*, vol. 20, no. 3, pp. 3-6, 2020.

[25] W. Wang, L. Wang, P. Zhang, S. Xu, K. Fu, L. Song, S. Hu, "A privacy protection scheme for telemedicine diagnosis based on double blockchain," *Journal of Information Security and Applications*, vol. 61, no. 2, pp. 1-12, 2021.

Biography

Yajie Zhang is a lecturer in Law Teaching and Research Department of the Party School of the CPC Shijiazhuang Municipal Committee. She graduated from Hebei University of Economics and Business. Her research directions include civil and commercial law and administrative law.

Sparrow Search Algorithm Based on Adaptive Weighting and Fusion of Inverse and Local Learning

Yehua Chen[1], Junfu Xi[2], Xia Liu[3], and Huan Gao[2]
(Corresponding author: Huan Gao)

Accounting Department, Hebei Vocational University of Technology and Engineering[1]
Xingtai, Hebei 054035, China
Information Engineering Department, Hebei Vocational University of Technology and Engineering[2]
Xingtai, Hebei 054035, China
Email: 393888757@qq.com
School of Mathematics and Information Technology, Xingtai University[3]
Xingtai, Hebei 054000, China

(Received Oct. 9, 2022; Revised and Accepted Apr. 11, 2023; First Online Apr. 27, 2023)

Abstract

Aiming at the problems of slow convergence speed, low convergence accuracy, and reduced population diversity in the late optimization stage of the sparrow search algorithm (SSA), an adaptive weight and fusion of reverse and local learning-based sparrow search algorithm (ARLSSA) is proposed. First, inspired by the particle swarm optimization algorithm, an adaptive inertia weight factor is designed and introduced into the finder position update to balance the algorithm's global and local search capabilities. Then a fusion strategy of reverse and local learning is designed to enhance population diversity, strengthen searchability and improve the ability to jump out of the local optimum. Finally, six algorithms are tested and compared in different dimensions through 12 benchmark functions. The experimental results show that the ARLSSA has higher solution accuracy, convergence speed, and stability than the other five algorithms.

Keywords: Adaptive Weights; Local Iterative; Reverse Learning; Sparrow Search Algorithm

1 Introduction

Global optimization problems have been widely used in financial engineering, national defense military, production scheduling, and engineering design. Its mathematical model can be described as:

$$\min_{x} f(x),\ x = (x_1, x_2, \ldots, x_n)^T \in R^n \tag{1}$$

where $x_i \in [l_i, u_i]$, $(i = 1, 2, \ldots, n)$, l_i and u_i are the boundaries of x_i.

For solving high-dimensional complex global optimization problems, swarm intelligence optimization algorithms have obvious advantages over traditional optimization methods, and have greatly attracted the attention in related fields [6, 8, 9]. A variety of algorithms based on swarm intelligence optimization, such as particle swarm optimization (PSO) [7], butterfly optimization algorithm (BOA) [3], cuckoo algorithm (CS) [22], differential evolution algorithm (DE) [2], whale optimization Algorithm (WOA) [13], and Grey Wolf Optimization Algorithm (GWO) [14], have been proposed to solve high-dimensional complex global optimization problems.

Sparrow search algorithm (SSA) [19] is a new swarm intelligence optimization algorithm proposed by Xue et al. in 2020. This algorithm is derived from the simulation of the foraging behavior and anti-predation behavior of sparrows in nature. It realizes the search and optimization through the mutual division of labor among the finder-joiner-warner in the sparrow population, and has the characteristics of less adjustment parameters and strong optimization performance. It has been successfully applied to parameter identification of industrial robot [20], Lidar echo decomposition [24], feature extraction of new coronary pneumonia images [25], optimization of refrigeration unit load [23], support vector machine model parameter optimization [17], battery stack parameter optimization identification [26], and UAV trajectory planning problem [16].

However, the sparrow search algorithm still has the disadvantages of dependence on the initial solution, sudden decline in population diversity in the later iteration, and easy to fall into local optimum, especially when solving complex high-dimensional optimization problems. To overcome these shortcomings, many different methods have been proposed to improve the optimization of SSA, which are divided into three categories: 1) Change the initialization method of population. Since the random initialization of the population in SSA cannot guarantee better population diversity, a reverse learning method in the initialization stage of the sparrow population was introduced to enhance the diversity [10]. The Tent mixed pure sequence was introduced to initialize the population, improving the quality of the initial solution and enhancing the global search ability of the algorithm [12]; 2) Modify the position update equations. In SSA, the finders guide the population to search and forage, and the position update equation provides a great help to the algorithm convergence [11]. In the late search stage, precocious convergence is prone to occur. When joiners move to the optimal position, the population diversity is prone to appear, making the algorithm easy to fall into local optimum. Combined with the idea of flight behavior in the bird flocking algorithm, the position update method of finder-joiner is improved to enhance the global search ability and local development ability of the algorithm [5]. Combined with the random following strategy of the chicken swarm algorithm, the position update method of joiners in the algorithm is improved. The global search and local development capabilities of the algorithm are balanced; 3) Introduce a new mechanism. Different mechanisms have different search capabilities. The introduction of new mechanisms to improve the optimization performance of SSA is also an improved research direction. Ouyang et al. proposed a lens learning sparrow search algorithm [15], which introduced lens reverse learning, variable spiral search strategy and annealing algorithm to effectively improve the search ability and the ability to jump out of local optimum. Yang et al. introduced a variable helical factor mechanism and an improved iterative local search strategy to improve the search accuracy and the ability to jump out of local optimum [21]. Wu et al. proposed an improved circle mixed pure mapping theory and combined with quantum mechanism to randomly initialize the population [18]. An enhanced search strategy and new boundary control strategy were used to improve the optimization ability of the algorithm.

Even though some improvements have been made based on the standard SSA, there are still some shortcomings:

1) In the optimization process of SSA, the individuals who have completed the optimization will form their own experience; especially the position update formula of finders ignores the learning of the position information of the best individuals in the history of the current population.

2) The existing literatures have not fundamentally changed the SSA optimization mechanism, lack learning ability, and may still fall into local optimum when encountering high-dimensional complex problems.

To solve the above problems, a sparrow search algorithm based on adaptive weight and fusion of reverse and local learning (ARLSSA) is proposed. An adaptive weighting strategy is introduced to adaptively adjust the finder position update through the distribution of the historical optimal position of the current population in the dimension, so as to improve the convergence speed and accuracy of the algorithm. Through fusing the reverse and local learning strategy, the learning ability and the ability to jump out of local optimum are improved to increase the population diversity. To verify the optimization of this algorithm, ARLSSA, PSO, GWO, WOA, SSA, CSSOA [12] are tested and analyzed on 12 benchmark functions. The results show that ARLSSA algorithm is effective and feasible in convergence accuracy, stability and convergence speed. The main contributions of this paper are as follows:

1) According to the distribution of the historical optimal positions of the population in the D dimension, a method of adaptively adjusting the weights is designed to balance the global search and local development capabilities of the algorithm.

2) A fusion of reverse and local learning strategy is proposed to improve the ability of the population to jump out of local optimum when dealing with high-dimensional complex problems.

3) The effectiveness of the ARLSSA algorithm is verified by using the benchmark functions.

The main arrangements of this paper are as follows: Section 2 introduces the standard sparrow search algorithm; Section 3 introduces and analyzes the ARLSSA algorithm; Section 4 gives the experimental parameters, environment and verification for the effectiveness of the ARLSSA algorithm, and Section 5 draws the conclusions.

2 Sparrow Search Algorithm

The sparrow search algorithm realizes the purpose of search and optimization through the mutual division of labor between finder-joiner-warner. Finders are responsible for food in the population and provide foraging areas and directions for the entire sparrow population, generally accounting for 10%-20% of the total population. Joiners use finders to obtain food. Warners are randomly selected individuals in the population, accounting for about 10%-20% of the total population. When they are aware of danger, they will issue an alarm to let the sparrows escape to a safe area.

At each iteration of SSA, the position update formula of finders is:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^{t} \cdot \exp\left(\frac{-i}{a \cdot T}\right), & \text{if } R_2 < ST \\ X_{i,j}^{t} + Q \cdot L, & \text{if } R_2 \geq ST \end{cases} \tag{2}$$

where t is the number of current iterations; T is the maximum number of iterations; $X_{i,j}$ is the position of the i-th sparrow in the j-th dimension; $a \in (0,1]$ is a random number; $R_2 \in [0,1]$ and $ST \in [0.5,1]$ are the early warning value and safety threshold, respectively; Q is a random number subject to standard normal distribution; L is a matrix with a size of $1 \times D$ and elements of 1, and D is a population dimension. $R_2 < ST$ means that no predators are found around the foraging environment, and the finders can perform extensive searches to guide the population to obtain a higher fitness. $R_2 \geq ST$ indicates that predators are found around the foraging environment, and the finders adjust the search strategy to escape and lead the population to a safe position.

The position update formula of joiners is as follows:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_W^t - X_{i,j}^t}{i^2}\right), & \text{if } i > \frac{N}{2} \\ X_P^{t+1} + \left|X_{i,j}^t - X_P^{t+1}\right| \cdot A^+ \cdot L, & \text{otherwise} \end{cases} \tag{3}$$

where X_W is the current worst position; X_P is the optimal position currently occupied by the finders; A is a matrix with $1 \times D$, and each element is randomly assigned to 1 or -1, and $A^+ = A^T \left(AA^T\right)^{-1}$; N is the population size; $i > N/2$ indicates that the i-th joiner with poor fitness does not get food and needs to fly elsewhere for food; otherwise, the i-th joiner forages around the finders in the best position.

The position update formula of warners is as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_B^t + \beta \cdot \left|X_{i,j}^t - X_B^t\right|, & \text{if } f_i > f_g \\ X_{i,j}^t + K \cdot \left(\frac{\left|X_{i,j}^t - X_W^t\right|}{(f_i - f_\omega) + \varepsilon}\right), & \text{if } f_i = f_g \end{cases} \tag{4}$$

where $X_B$is the current global optimal position; β is a compensation control parameter, obeying a normal distribution random number with mean 0 and variance 1; $K \in [-1,1]$ is a random number; ε is a minimum constant; f_i, f_g and f_ω are the fitness values of the i-th sparrow, and the optimal and worst fitness values of the current sparrow population respectively. $f_i > f_g$ indicates that the i-th sparrow is in the marginal area of the population and vulnerable to predators; $f_i = f_g$ indicates that the first sparrow in the middle of the population is aware of the danger and needs to escape from its current position.

3 Sparrow Search Algorithm Based on Adaptive Weight and Fusion of Inverse and Local Learning

3.1 Adaptive Weight Strategy

In SSA, finders guide the population to search and forage. From Equation (2), when $R_2 < ST$, $\exp\left(\frac{-i}{a \cdot T}\right)$ adopts a linear decreasing inertia weight strategy. That is, the value of the inertia weight ϖ decreases linearly from a larger value in the early stage to a smaller value at each iteration. This aims to ensure that the algorithm has better global search and local development capabilities in the later stage. According to the parameters of i, a and T and set values,, it can be known that the larger value of ϖ in the early stage of SSA is usually close to constant 1. When $R_2 \geq ST$, ϖ is 1. A larger ϖ can facilitate global search and increase the population diversity; a smaller ϖ can improve the local mining ability and speed up the convergence speed of the algorithm [4]. Many scholars have pointed out that in the process of groups, such as birds and fish searching for their own food h, the individuals who have completed the search would form their own experience, especially the experience of historical optimal individuals are more worth learning. However, the position update formula of finders ignores the learning of the position information of the optimal individuals in the history of the current population. To solve these problems, this paper proposes a method of adaptively adjusting the weights according to the distribution of the historical optimal position of the current population in the D dimension. The formula is as follows:

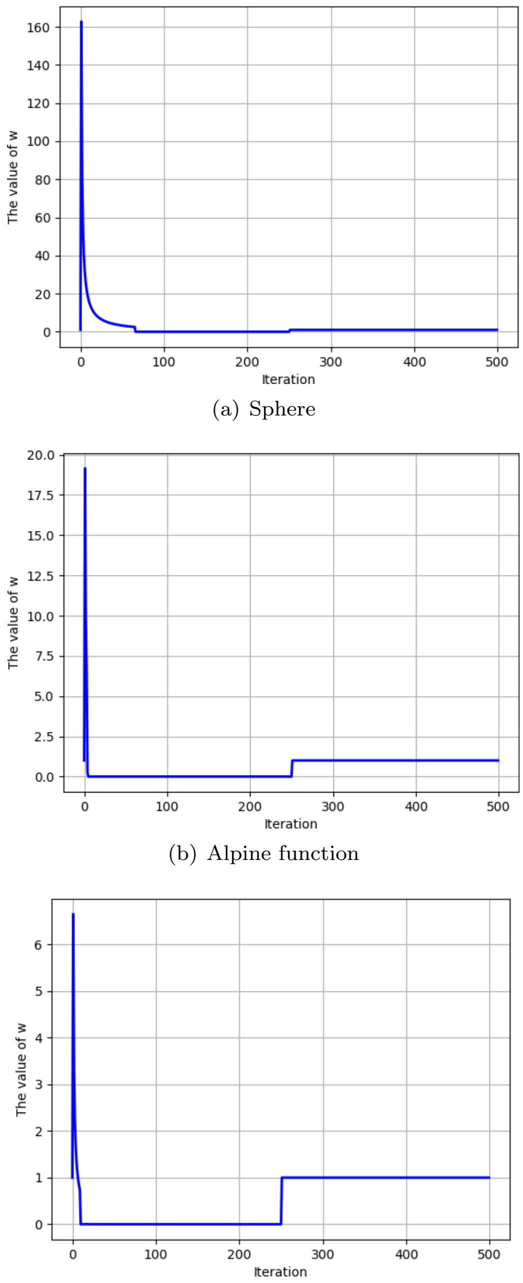
$$\omega(t+1) = \begin{cases} \left(p_w - p_b\right)/t, & \text{if } t \leq T/b,\ t \neq 0 \\ 1, & \text{otherwise} \end{cases} \tag{5}$$

where ω is the inertia weight, and its initial value is 1; p_w and p_b are respectively the maximum value and minimum value of the historical optimal position of the population in D dimension at the i-th iteration; b is a positive number and the default value is 2. $t \leq T/2$ indicates that that the finders adopt an adaptive weight adjustment method to guide the population to search and forage. Otherwise, it means that the population is in short supply of food in the foraging environment, and the finders adjust the weight strategy for foraging.

The proposed adaptive weight adjustment method is introduced into the finder position update formula, and the improved formula is as follows:

$$X_{i,j}^{t+1} = \begin{cases} \omega(t+1) \cdot X_{i,j}^t \cdot \exp\left(\frac{-i}{a \cdot T}\right), & \text{if } R_2 < ST \\ \omega(t+1) \cdot X_{i,j}^t + Q \cdot L, & \text{if } R_2 \geq ST \end{cases} \tag{6}$$

Several different types of test functions are tested Limited by space, Figure 1 shows the inertia weight ω curve of the SSA solution benchmark test functions (Sphere, Alpine and Easom) with adaptively adjusted inertia weights. Sphere is the high-dimensional unimodal function; Alpine is the high-dimensional multimodal function, and Easom is a fixed 2-dimensional function. The high-dimensional function dimension is set to $D = 30$, the population size $N = 30$; the maximum number of iterations is $T = 500$, and the fixed-dimensional function dimension is set to $D = 2$.



(a) Sphere

(b) Alpine function

(c) Easom function

Figure 1: Adaptive adjustment inertia weight ω curve

As shown in Figure 1, when optimizing three different test functions, SSA with adaptive adjustment of inertia weights obtains a large ω value at the beginning of iteration. This avoids the algorithm falling into the defect of small range search at the beginning of iteration and is conducive to global search. As the iterations of the sparrow population increase, ω will gradually decrease, which is conducive to the local detailed search of the algorithm.

3.2 Fusion of Reverse and Local Learning Strategy

In the optimization process, like other swarm intelligence optimization algorithms, SSA has a contradiction between group diversity and algorithm convergence speed. For the improvement of standard SSA, whether it is changing the population initialization method, modifying the position update formula or introducing a new mechanism, its aim is to improve the local search ability while maintaining the population diversity and preventing premature convergence while the algorithm converging rapidly. The SSA based on the fusion strategy of reverse and local learning is also generated based on this idea. That is, relying on the reverse learning mechanism to make multiple sparrow individuals learn reversely. While enhancing the population diversity, the local mining capacity of the historical optimal individuals in the population is improved through the local iterative search mechanisms. The specific improvement strategies are as follows:

1) Inverse learning of sparrows. The reverse learning strategy can improve the search ability of swarm optimization algorithm [1]. When it is detected that the historical optimal position has not been updated within a certain number of iterations, it is determined that the algorithm has fallen into a stagnant state and the finders cannot find a better solution in their fields. To this end, the sparrows in the $n$optimal historical positions in the current joiners to perform the reverse learning of the c-th generation, making them search in a larger range of activities and improve the success rate. The learning methods of other $N-n$ sparrows are constant. Specific steps are as follows:

Step 1. The i-th sparrow of reverse learning is $x_i^j = (x_i^1, x_i^2, \ldots, x_i^D)$ $(i = 1, 2, \ldots, n)$ $(j = 1, 2, \ldots, D)$, and its reverse solution $\bar{x}_i^j = (\bar{x}_i^1, \bar{x}_i^2, \ldots, \bar{x}_i^D)$ can be defined as follows:

$$\bar{x}_i^j = k \cdot (l^j + u^j) - x_i^j \quad (7)$$

where $k \in (0, 1)$ is a random number subject to uniform distribution; $x_i^j \in [l^j, u^j]$ and $[l^j, u^j]$ are the dynamic boundaries of the j-th dimensional search space, $l^j = \min(x_i^j)$ and $u^j = \max(x_i^j)$.

Step 2. If $\bar{x}_i^j$ appears and crosses the dynamic boundary $[l^j, u^j]$ to be a non-feasible solution, it is reset by the following formula:

$$\bar{x}_i^j = \text{rand}(l^j, u^j) \quad (8)$$

Step 3. When the reverse learning of generation c is completed, the reverse subpopulation and the subpopulation of the joiners in the generation c are merged in the order of original individual position and participate in the later evolutionary competition. The optimization process of generation $c+1$ restores the search mechanism before reverse learning.

2) Partial learning of sparrows. The local iterative search of the historical optimal position neighborhood in the population can effectively prevent the

sparrow individual from missing a better solution in the process of jumping directly to the current optimal position. At the same time, local search near the current optimal solution helps to improve the accuracy of the solution and jump out of local optimum in a small range. Specific steps are as follows:

Step 1. After the sparrow population is updated at each generation, the historical optimal position x^* of the population is updated, and x^* is disturbed to obtain the intermediate solution x^{**}. The update formula is as follows:

$$x^{**} = x^* \cdot \mathrm{rand}() \tag{9}$$

where rand() is a random number between 0 and 1.

Step 2. Boundary detection is performed on x^{**}, if it crosses, return to *Step 1*; otherwise the fitness value $f(x^{**})$ of the intermediate solution x^{**} is calculated;

Step 3. For the local iterative search result x^{**}, the greedy retention strategy is adopted, that is,

$$\begin{aligned} x^* &= \begin{cases} x^{**}, & f(x^{**}) < f(x^*) \\ x^*, & f(x^{**}) \geq f(x^*) \end{cases} \\ f(x^*) &= \begin{cases} f(x^{**}), & f(x^{**}) < f(x^*) \\ f(x^*), & f(x^{**}) \geq f(x^*) \end{cases} \end{aligned} \tag{10}$$

3.3 RLSSA Pseudo Code

The pseudo code of ARLSSA is shown in Algorithm 1.

3.4 Complexity Analysis of ARLSSA Algorithm

N is the sparrow population; D is the dimension; T is the maximum number of iterations; s_1 is the randomly initialized population time; j_d is the time to solve the individual fitness value; s_2 is the time to sort the population according to the fitness value; Pd is the number of finders; s_3 is the update time of each dimension; s_4 is the update time of joiners in each dimension; Sd is the number of warners, and s_5 is the update time of each dimension. $TC_1 = O(s_1 + N \times j_d + s_2)$ is the time complexity in the initial stage; $TC_3 = O((N - Pd) \times s_4 \times D)$ is the time complexity of finder position update; $TC_3 = O((N - Pd) \times s_4 \times D)$ is the time complexity of joiner position update; $TC_4 = O(Pd \times s_5 \times D)$ is the time complexity of warner position update, and $TC_5 = O(N \times j_d + s_2)$ is the time complexity of calculating the fitness value and sorting of sparrows in the t-th generation. In this way, the time complexity of SSA is $TC = TC_1 + (TC_2 + TC_3 + TC_4 + TC_5) \times T = O(D + j_d)$.

In ARLSSA, s_{11} is the time for local learning; s_{12} is the learning time for reverse learning in each dimension,

Algorithm 1 ARLSSA algorithm

```
Input: the search space dimension $D$ of the algorithm, the population size $N$, the maximum number of iterations $T$, the upper and lower bounds of initial value $lb$ and $ub$, the number of finders $Pd$, the safe threshold $ST$, the value of $\varepsilon$, the number of warners $Sd$, the adaptive adjustment factor $b$, the reverse learning algebra $L_t$, and the subpopulation size of reverse learning $n$;
Output: the historical optimal solution $X_B$of the population.
Begin
According to $D$ and $N$, the population is randomly initialized in the search space;
The fitness value of each sparrow individual $f(x_i)$ is calculated, $i = 1, 2, \ldots, N$, and sorted according to the fitness value and its corresponding position. The values of $f_g$, $f_\omega$, $X_B$ and $X_W$ are recorded, let $t = 0$;
while $t < T$ do
  From the sparrow population, the individuals whose fitness values are $Pd * N$ are selected as finders and the positions are updated according to formula (6);
  if satisfies the reverse learning condition then
    According to the subpopulation size $n$ of the reverse learning, the reverse learning of joiners is carried out according to formulas (7) and (8);
    The remaining sparrow individuals after $(1-Pd) \cdot N - n$ are selected as joiners, and position update is carried out according to formula (3);
  else
    The remaining sparrow individuals after $(1-Pd) \cdot N$ are selected as joiners, and the position is updated according to formula (3);
  end if
  The individuals with $Sd * N$ are randomly selected from the sparrow population as early warners and the position is updated according to formula (4);
  The fitness value is calculated and sorted according to the fitness value and its corresponding position to update the values of $f_g$, $f_\omega$, $X_B$ and $X_W$;
  Local learning on $X_B$ is performed according to formula (9); the values of $f_g$ and $X_B$ are updated according to formula (10), and the values of $p_w$ and $p_b$ are recorded to solve the inertia weight value $\omega$ according to formula (5);
  $t = t + 1$;
end while
End
```

and s_{13} is the time for inertia weight calculation. $TC_{11} = O(s_1 + N \times j_d + s_2)$ is the time complexity in the initial stage; $TC_{22} = O(Pd \times s_3 \times D)$ is the time complexity of finder position update; $TC_{33} = O(((N - Pd - N/L_t) \times s_4 + N/L_t \times s_{12}) \times D)$ is the time complexity of joiner position update; $TC_{44} = O(Pd \times s_5 \times D)$ is the time complexity of warner position update; $TC_{55} = O(N \times j_d + s_2)$ is the time complexity of calculating the sparrow fitness value and sorting in the t-th generation; $TC_{66} = O(s_{11})$ is the time complexity of local learning in the t-th generation, and $TC_{77} = O(s_{13})$ is the time complexity of calculating the inertia weight value. The time complexity of ARLSSA is $TC_0 = TC_{11} + (TC_{22} + TC_{33} + TC_{44} + TC_{55} + TC_{66} + TC_{77}) \times T = O(D + j_d)$. $TC = TC_0$ indicates that the time complexity of ARLSSA and SSA algorithms are consistent.

4 Experiment and Analysis

4.1 Parameter Settings and Test Functions

To verify the performance of ARLSSA 12 commonly used benchmark functions are selected to test and compare PSO, GWO, WOA, SSA, CSSOA and ARLSSA. The specific parameter settings are all derived from the original literature as shown in Table 1, and the benchmark functions are shown in Table 2. $F_1 \sim F_5$ is the high-dimensional unimodal function; $F_6 \sim F_{10}$ is the high-dimensional multimodal function, and F_{11} and F_{12} are the fixed dimension functions ($D = 2$). The experimental environment is based on Intel(R) Core(TM) i7-9700 CPU@3.00GHz, memory 16GB, Windows 10 64-bit operating system with Python 3.9.1 installed.

4.2 Convergence Accuracy and Stability Analysis

For fairness, the population size and the maximum number of iterations of each algorithm are set to 30 and 500, respectively. Based on the benchmark function dimensions set to 30, 50, and 100 respectively, each algorithm runs independently 30 times. Their optimal value (*Best*), mean (*Ave*) and standard deviation (*Std*) are selected as evaluation indicators. The mean and the standard deviation respectively represent the convergence accuracy and the stability of the algorithm. The optimal results are shown in Table 3 (in bold).

As shown in Table 3, in the five high-dimensional unimodal functions $F_1 \sim F_5$, when the dimensions $D = 30$, $D = 50$ and $D = 100$ are set for optimization, the optimal value (*Best*), the average value (*Ave*) and the standard deviation (*Std*) of the six algorithms indicate that ARLSSA found the theoretical optimal value 0 when solving the three functions of F_1, F_2 and F_3. The obtained *Ave* and *Std* are also 0. In addition, SSA and CSSOA also found the theoretical optimal value 0 when solving F_1, F_2 and F_3 in $D = 30$ and $D = 50$. The *Ave* and *Std* of ARLSSA optimization results are much higher than the other five algorithms. Then, when solving F_4, the *Best*, *Ave* and *Std* of ARLSSA are at least 3 orders of magnitude higher than the other 5 calculations. When solving F_5, the *Best*, *Ave* and *Std* of ARLSSA are at least 1 order of magnitude higher than the other five calculations.

In the five high-dimensional multimodal functions $F_6 \sim F_{10}$, when $D = 30$, $D = 50$ and $D = 100$ are set for optimization, ARLSSA finds the theoretical optimal value 0 when solving F_6, F_8, F_9 and F_{10}. The obtained *Ave* and *Std* are also 0. Only when solving F_6 and F_8, the *Best*, *Ave* and *Std* of CSSOA are 0, and the optimization performance of the other four algorithms decreases as the dimension increases. For the solution of F_7, ARLSSA, SSA and CSSOA have basically the same optimization ability, and ARLSSA does not reflect the superiority of the algorithm.

In the two fixed dimension functions F_{11} and F_{12}, ARLSSA found the theoretical optimal value when solving F_{11} and F_{12}, and *Std* is also 0. PSO also found the theoretical optimal values when solving F_{11} and F_{12}. For the solution of F_{12}, the other four algorithms also obtained theoretical optimal values.

The optimization results of the six algorithms in different dimensions show that compared with the other five algorithms, whether it is a high-dimensional unimodal, high-dimensional multimodal or a fixed-dimensional function, ARLSS not only has significantly improved the optimization accuracy, but also has better stability.

4.3 Convergence Curve Analysis

To intuitively compare the optimization process and convergence speed of the six algorithms, Figures 2–13 show the convergence curves of the six algorithms in the dimension $D = 50$ of 12 benchmark test functions.

It can be seen from the above 12 figures, ARLSSA has fast convergence speed and high convergence accuracy in the functions $F_1 \sim F_3$, F_6 and $F_8 \sim F_{10}$. It has strong anti-local attraction ability, excellent search ability, and the optimization effect is much higher than other algorithms. ARLSSA also has a good convergence effect in the functions F_4 and F_5. Its optimization effect is better than other algorithms, but it does not jump out after falling into a local extreme point. ARLSSA and CSSOA have basically the same convergence speed in the functions F_7 and F_{11}. ARLSSA does not reflect the superiority of the algorithm, but its convergence accuracy is higher than CSSOA. ARLSSA does not converge as fast as the other four algorithms in the function F_{12}. Through the above analysis, it can be seen that the ARLSSA algorithm gets rid of the limitation of the original algorithm search mechanism and improves the solution efficiency and quality of the algorithm.

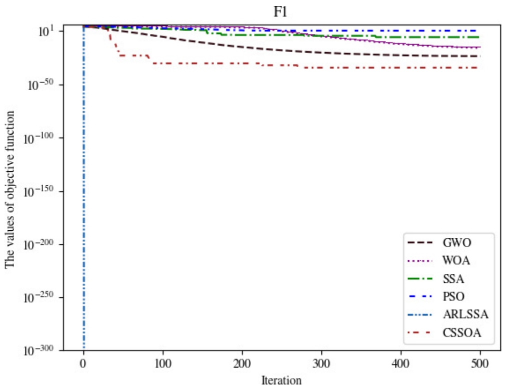


Figure 2: $d = 50$, Convergence curve of $F_1(x)$

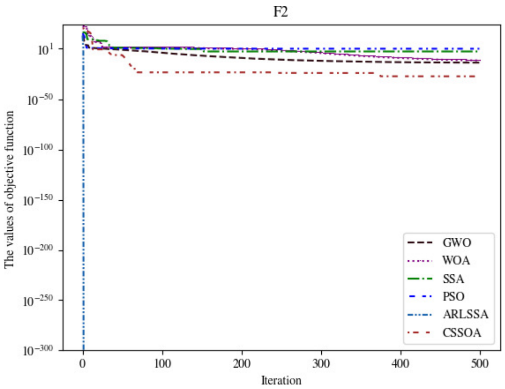


Figure 3: $d = 50$, Convergence curve of $F_2(x)$

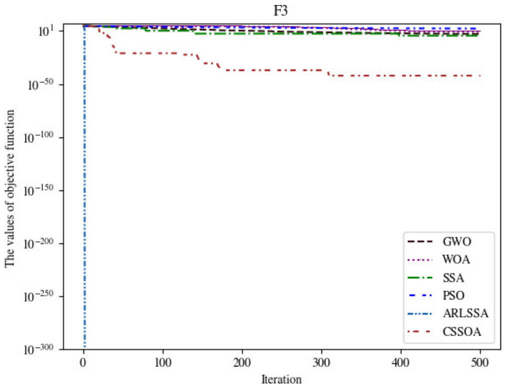


Figure 4: $d = 50$, Convergence curve of $F_3(x)$

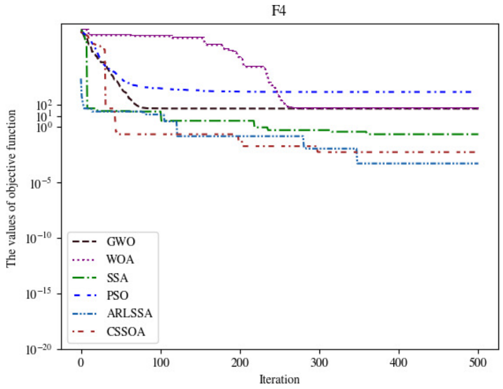


Figure 5: $d = 50$, Convergence curve of $F_4(x)$

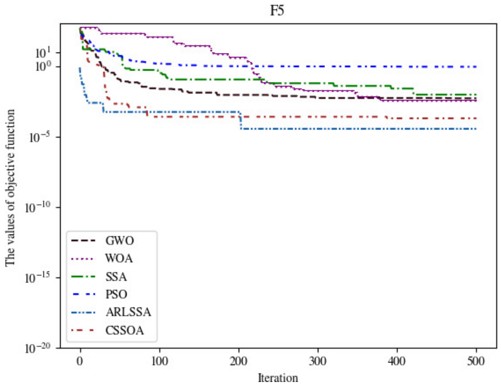


Figure 6: $d = 50$, Convergence curve of $F_5(x)$

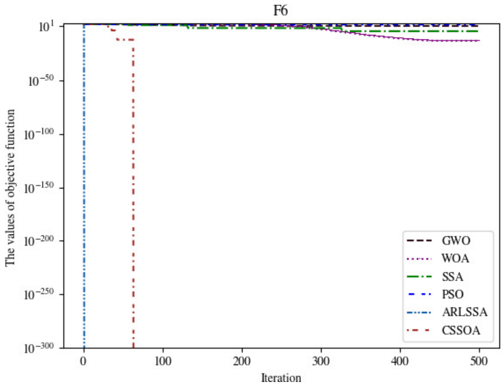


Figure 7: $d = 50$, Convergence curve of $F_6(x)$

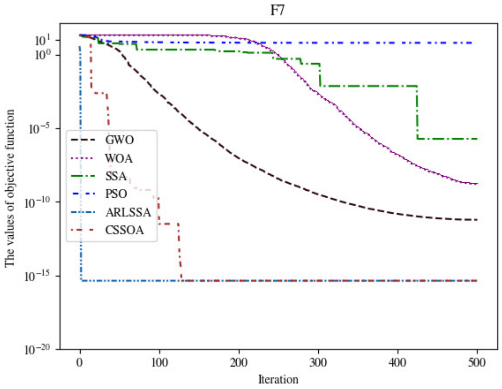


Figure 8: $d = 50$, Convergence curve of $F_7(x)$

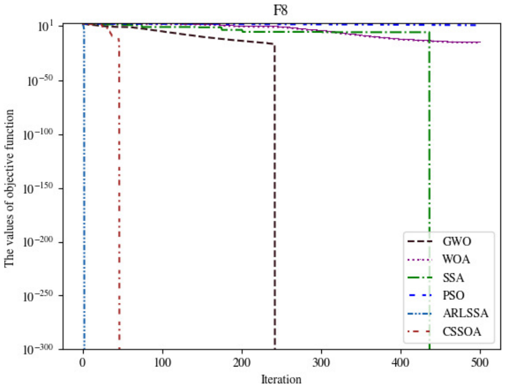


Figure 9: $d = 50$, Convergence curve of $F_8(x)$

Table 1: Parameters

Algorithm	PSO	GWO	WOA	SSA	CSSOA	ARLSSA
Parameter	$c_1 = 2$ $c_2 = 2$ $W_{\min} = 0.2$ $W_{\max} = 0.9$	$a = (2 \to 0)$	$b = 1$	$ST = 0.8$ $Pd = 0.2$ $Sd = 0.2$	$ST = 0.8$ $Pd = 0.2$ $Sd = 0.2$	$ST = 0.8$ $Pd = 0.2$ $Sd = 0.2$ $L_t = 20$ $n = 0.1$

Table 2: Test functions

Function	Formula	Domain	Min
Sphere	$F_1(x) = \sum_{i=1}^{n} x_i^2$	$[-100, 100]$	0
Schwefel's	$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	$[-10, 10]$	0
Quadric	$F_3(x) = \sum_{i=1}^{n} \sum_{j=1}^{i} x_j^2$	$[-100, 100]$	0
Rosenbrock	$F_4(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-30, 30]$	0
Quartic	$F_5(x) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0, 1]$	$[-1.28, 1.28]$	0
Rastrigin	$F_6(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	$[-5.12, 5.12]$	0
Ackley	$F_7(x) = -20\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i)\right) + 20 + e$	$[-32, 32]$	0
Griewank	$F_8(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]$	0
Zakharov	$F_9(x) = \sum_{i=1}^{n} x_i^2 + \left(\sum_{i=1}^{n} 0.5 i x_i\right)^2 + \left(\sum_{i=1}^{n} 0.5 i x_i\right)^4$	$[-5, 10]$	0
Alpine	$F_{10}(x) = \sum_{i=1}^{n} \lvert x_i \sin(x_i) + 0.1 x_i \rvert$	$[-10, 10]$	0
Easom	$F_{11}(x) = -\cos(x_1)\cos(x_2)\exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$	$[-100, 100]$	-1
Six-Hump Camel	$F_{12}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	$[-5, 5]$	-1.0316

5 Conclusion

Based on the standard sparrow search algorithm, an adaptive inertia weight and a fusion strategy of reverse and local learning are introduced. A sparrow search algorithm based on an adaptive weight and a fusion of reverse and local learning is proposed. First, an adaptive inertia weight factor is introduced into the finder position update formula, so that the algorithm has strong global search ability in the early iteration, and a high local mining ability in the later iteration. Then a reverse and local learning strategy is integrated to increase the population diversity, improve the search ability and the ability to jump out of local optimum. Finally, 12 benchmark functions are selected for comparison with other five algorithms in different dimensions of functions. The experimental results show that the ARLSSA algorithm has high convergence accuracy, convergence speed and solution stability, and strong competitiveness. Our next work would apply ARLSSA to practical engineering problems, such as image segmentation and face recognition, to further test its effectiveness.

Acknowledgments

The work was supported by Xingtai Science and Technology Planning Project (2020ZZ048).

References

[1] N. A. Alawad and B. H. Abed-alguni, "Discrete island-based cuckoo search with highly disruptive polynomial mutation and opposition-based learning strategy for scheduling of workflow applications in cloud environments," *Arabian Journal for Science*

Table 3: Comparison of optimal performance of 6 algorithms

Functions	Dim	Index	PSO	GWO	WOA	SSA	CSSOA	ARLSSA
F_1	$D = 30$	Best	4.36E-01	8.21E-32	1.55E-22	**0**	**0**	**0**
		Ave	1.472121	2.64E-31	6.56E-20	4.00E-09	1.53E-48	**0**
		Std	7.28E-01	3.78E-31	1.51E-19	5.65E-09	2.17E-48	**0**
	$D = 50$	Best	1.39E+01	1.68E-22	1.14E-17	**0**	**0**	**0**
		Ave	1.30E+01	2.34E-22	1.32E-17	1.57E-05	9.13E-46	**0**
		Std	4.566992	9.23E-23	2.52E-18	2.22E-05	1.29E-45	**0**
	$D = 100$	Best	1.03E+02	3.22E-14	2.41E-13	**0**	**0**	**0**
		Ave	1.21E+02	5.05E-14	1.72E-12	1.03E-04	5.70-42	**0**
		Std	2.54E+01	2.58E-14	2.10E-12	3.43E-04	3.6142	**0**
F_2	$D = 30$	Best	1.189437	1.24E-19	1.94E-15	**0**	**0**	**0**
		Ave	4.633723	7.93E-19	7.53E-14	1.95E-02	1.32E-18	**0**
		Std	1.540803	5.46E-19	1.80E-13	5.86E-02	7.27E-18	**0**
	$D = 50$	Best	2.397996	2.41E-14	2.25E-12	**0**	**0**	**0**
		Ave	1.38E+01	8.90E-14	2.60E-11	0.30E-01	1.71E-16	**0**
		Std	2.827711	5.04E-14	3.60E-11	5.84E-02	7.65E-16	**0**
	$D = 100$	Best	4.20E+01	2.74E-09	2.26E-10	3.68E-30	8.35E-92	**0**
		Ave	4.96E+01	4.37E-09	2.76E-09	4.52E-01	1.03E-19	**0**
		Std	6.745085	1.28E-09	2.94E-09	8.05E-01	4.20E-18	**0**
F_3	$D = 30$	Best	3.83E+01	1.30E-08	1.26E-04	**0**	**0**	**0**
		Ave	9.45E+01	3.10E-06	3.56E-02	1.45E-05	2.17E-30	**0**
		Std	6.67E+01	6.71E-06	8.42E-02	4.59E-05	6.86E-30	**0**
	$D = 50$	Best	5.83E+02	8.55E-04	1.70E-02	**0**	**0**	**0**
		Ave	1.32E+03	3.34E-01	8.70E-01	2.02E-02	7.08E-43	**0**
		Std	6.24E+02	5.83E-01	1.516248	3.70E-02	2.06E-42	**0**
	$D = 100$	Best	6.70E+03	1.41E+02	2.57E+01	1.21E-07	2.30E-70	**0**
		Ave	1.88E+04	7.53E+02	6.29E+02	2.50E-04	2.05E-40	**0**
		Std	1.07E+04	7.86E+02	1.13E+03	3.54E-04	2.90E-40	**0**
F_4	$D = 30$	Best	6.36E+01	2.60E+01	2.67E+01	3.41E-02	3.14E-03	**1.04E-06**
		Ave	1.72E+02	2.70E+01	2.73E+01	1.109601	3.61E-02	**5.63E-05**
		Std	8.19E+01	5.06E-01	6.90E-01	1.475085	6.60E-02	**6.68E-05**
	$D = 50$	Best	8.38E+02	4.58E+01	4.69E+01	1.08E-01	3.96E-04	**4.94E-09**
		Ave	1.56E+03	4.70E+01	4.73E+01	8.60E-01	2.52E-02	**9.82E-06**
		Std	5.91E+02	6.14E-01	4.25E+01	9.48E-01	4.62E-02	**1.59E-05**
	$D = 100$	Best	1.57E+04	9.77E+01	9.65E+01	1.77E-02	1.05E-03	**2.42E-06**
		Ave	2.45E+04	9.82E+01	9.72E+01	1.488532	4.35E-02	**1.85E-05**
		Std	6.72E+04	3.19E-01	2.59E-01	1.665739	7.66E-01	**2.27E-05**
F_5	$D = 30$	Best	6.98E-02	7.75E-04	5.12E-05	7.35E-04	1.33E-05	**7.80E-06**
		Ave	2.01E-01	2.39E-03	5.93E-04	4.22E-03	8.51E-04	**1.30E-05**
		Std	9.93E-02	1.16E-03	5.37E-04	3.70E-03	6.22E-02	**1.18E-04**
	$D = 50$	Best	5.50E-01	1.51E-03	2.06E-05	3.66E-04	2.77E-05	**1.26E-06**
		Ave	1.047248	3.61E-03	2.01E-03	4.87E-03	1.61E-04	**2.73E-05**
		Std	1.047248	1.43E-03	2.50E-03	4.43E-03	1.65E-04	**1.00E-05**
	$D = 100$	Best	4.439801	4.85E-03	8.27E-05	5.56E-04	9.63E-05	**2.02E-06**
		Ave	9.014387	7.47E-03	1.56E-03	5.30E-03	8.56E-04	**1.51E-05**
		Std	3.970467	2.40E-03	1.33E-03	5.32E-03	6.30E-04	**1.42E-05**
F_6	$D = 30$	Best	2.86E+01	2.27E-13	**0**	**0**	**0**	**0**
		Ave	5.89E+01	3.560861	1.70E-14	2.50E-01	**0**	**0**
		Std	5.89E+01	3.580234	2.74E-14	5.42E-01	**0**	**0**
	$D = 50$	Best	1.21E+02	5.79E-12	**0**	**0**	**0**	**0**
		Ave	1.67E+02	5.147456	1.93E-13	2.85E-03	**0**	**0**
		Std	2.60E+01	3.491654	1.61E-13	8.90E-03	**0**	**0**
	$D = 100$	Best	4.93E+02	6.63E-10	1.59E-12	**0**	**0**	**0**
		Ave	6.29E+02	8.475884	2.72E-10	2.03E-01	**0**	**0**
		Std	1.49E+02	7.337488	7.80E-10	6.27E-01	**0**	**0**

F_7	$D = 30$	Best	3.252137	4.30E-14	3.11E-12	8.88E-16	8.88E-16	**4.44E-16**
		Ave	4.42E-03	6.29E-14	3.64E-11	4.99E-04	8.88E-16	**4.44E-16**
		Std	8.71E-01	9.20E-15	4.60E-11	1.54E-03	**0**	**0**
	$D = 50$	Best	4.959806	1.20E-12	1.26E-10	8.88E-16	8.88E-16	**4.44E-16**
		Ave	6.203646	2.06E-12	2.80E-09	2.09E-03	8.88E-16	**4.44E-16**
		Std	7.40E-01	8.19E-13	2.48E-09	3.51E-03	**0**	**0**
	$D = 100$	Best	8.148666	1.33E-08	7.16E-08	8.88E-16	8.88E-16	**4.44E-16**
		Ave	8.332857	1.84E-08	4.38E-07	9.92E-03	8.88E-16	**4.44E-16**
		Std	2.271.222	4.75E-09	6.37E-07	7.96E-03	**0**	**0**
F_8	$D = 30$	Best	2.48E+01	**0**	**0**	**0**	**0**	**0**
		Ave	3.04E+01	2.78E-03	2.84E-03	3.96E-05	**0**	**0**
		Std	2.952491	6.16E-03	8.99E-03	1.24E-04	**0**	**0**
	$D = 50$	Best	4.36E+01	**0**	**0**	**0**	**0**	**0**
		Ave	5.50E+01	3.90E-03	1.38E-03	1.78E-06	**0**	**0**
		Std	7.160443	8.25E-03	4.38E-03	5.48E-06	**0**	**0**
	$D = 100$	Best	1.07E+02	1.37E-14	3.79E-14	**0**	**0**	**0**
		Ave	1.17E+02	2.01E-03	1.12E-10	1.55E-06	**0**	**0**
		Std	8.219412	6.35E-03	3.01E-10	3.84E-06	**0**	**0**
F_9	$D = 30$	Best	2.15E+02	9.66E-11	1.52E-04	2.68E-169	8.13E-64	**0**
		Ave	5.68E+02	7.04E-08	5.73E-02	3.72E-04	8.89E-41	**0**
		Std	3.08E+02	2.11E-07	1.42E-01	4.40E-04	1.77E-40	**0**
	$D = 50$	Best	8.22E+02	1.92E-03	4.11E-02	6.82E-29	4.93E-60	**0**
		Ave	1.45E+03	3.88E-02	9.276504	7.96E-05	4.24E-37	**0**
		Std	4.32E+02	6.05E-02	1.71E+01	1.80E-05	1.30E-37	**0**
	$D = 100$	Best	2.72E+03	2.25E+01	1.05E+01	6.18E-54	9.01E-60	**0**
		Ave	8.95E+03	1.09E+02	5.12E+01	4.81E-01	4.97E-43	**0**
		Std	8.09E+03	6.83E+01	3.33E+01	9.67E-01	7.02E-42	**0**
F_{10}	$D = 30$	Best	5.29E-01	5.76E-14	1.35E-15	**0**	1.97E-65	**0**
		Ave	2.333099	4.61E-04	8.72E-06	2.62E-03	7.77E-20	**0**
		Std	1.643748	7.25E-04	2.75E-05	5.12E-03	2.45E-19	**0**
	$D = 50$	Best	4.819153	4.25E-10	7.12E-13	4.67E-65	3.82E-75	**0**
		Ave	6.428331	1.00E-03	8.48E-10	7.30E-03	5.71E-22	**0**
		Std	1.642993	7.80E-04	1.93E-09	1.55E-02	1.40E-21	**0**
	$D = 100$	Best	2.25E+01	1.50E-08	9.58E-10	5.26E-208	2.78E-50	**0**
		Ave	2.88E+01	3.06E-03	1.85E-05	1.14E-03	3.72E-24	**0**
		Std	4.796314	2.33E-03	5.85E-05	2.76E-03	7.89E-24	**0**
F_{11}	D=2	Best	**-1**	-0.999999	-0.99999	-0.99999	-0.99999	**-1**
		Ave	**-1**	-0.999999	-0.99999	-0.99999	-0.99999	**-1**
		Std	**0**	2.79E-07	3.47E-06	2.71E-06	8.57E-08	**0**
F_{12}	D=2	Best	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**
		Ave	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**	**-1.0316**
		Std	2.09E-16	6.72E-09	2.86E-08	8.71E-07	3.84E-08	**0**

and Engineering, vol. 46, pp. 3213–3233, November 2020.

[2] M. Z. Ali, N. H. Awad, and P. N. Suganthan, "Multi-population differential evolution with balanced ensemble of mutation strategies for large-scale global optimization," *Applied Soft Computing*, vol. 33, pp. 304–327, August 2015.

[3] S. Arora and S. Singh, "An effective hybrid butterfly optimization algorithm with artificial bee colony for numerical optimization," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no. 4, p. 14, 2017.

[4] S. Choudhary, S. Sugumaran, A. Belazi, and A. A. A. El-Latif, "Linearly decreasing inertia weight PSO and improved weight factor-based clustering algorithm for wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, October 2021.

[5] H. Fu and H. Liu, "Improved sparrow search algorithm with multi-strategy integration and its application," *Control and Decision*, vol. 37, pp. 87–96, January 2022.

[6] A. A. Heidari and P. Pahlavani, "An efficient modified grey wolf optimizer with lévy flight for optimization tasks," *Applied Soft Computing*, vol. 60, pp. 115–134, November 2017.

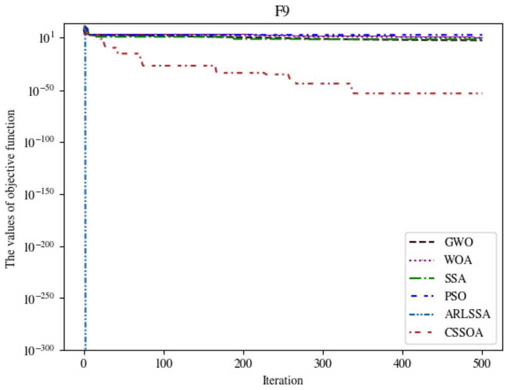


Figure 10: $d = 50$, Convergence curve of $F_9(x)$

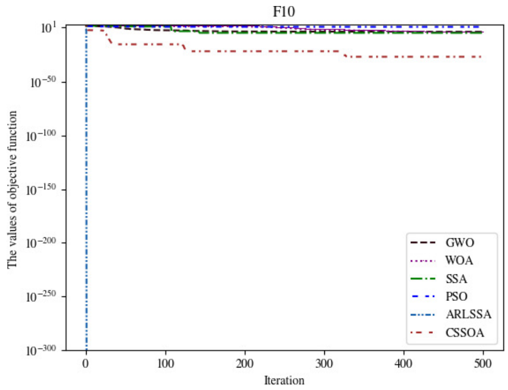


Figure 11: $d = 50$, Convergence curve of $F_{10}(x)$

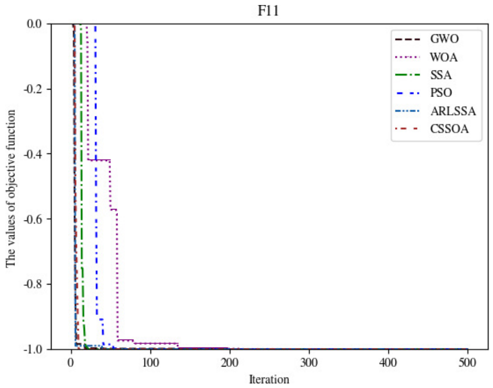


Figure 12: $d = 2$, Convergence curve of $F_{11}(x)$

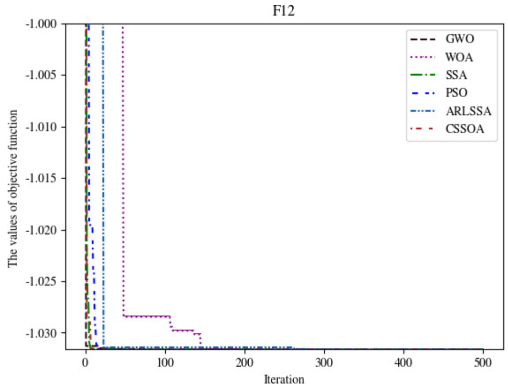


Figure 13: $d = 2$, Convergence curve of $F_{12}(x)$

[7] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, WA, Australia, 1995. IEEE.

[8] J. Liu, X. Liu, and Y. Li, "Two subpopulations cuckoo search algorithm based on mean evaluation method for function optimization problems," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 34, p. 2059027, November 2019.

[9] J. Liu, Y. Mao, X. Liu, and Y. Li, "A dynamic adaptive firefly algorithm with globally orientation," *Mathematics and Computers in Simulation*, vol. 174, pp. 76–101, August 2020.

[10] T. Liu, Z. Yuan, L. Wu, and B. Badami, "An optimal brain tumor detection by convolutional neural network and enhanced sparrow search algorithm," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 235, pp. 459–469, January 2021.

[11] X. Lyu, X. Mu, and J. Zhang, "Multi-threshold image segmentation based on improved sparrow search algorithm," *Systems Engineering and Electronics*, vol. 43, pp. 318–327, February 2021.

[12] X. Lyu, X. Mu, J. Zhang, and Z. Wang, "Chaos sparrow search optimization algorithm," *Journal of Beijing University of Aeronautics and Astronautics*, vol. 47, pp. 1712–1720, August 2021.

[13] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, May 2016.

[14] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, March 2014.

[15] C. Ouyang, D. Zhu, and Y. Qiu, "Lens learning sparrow search algorithm," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–17, May 2021.

[16] A. Tang, T. Han, D. Xu, and L. Xie, "Path planning method of unmanned aerial vehicle based on chaos sparrow search algorithm," *Journal of Computer Applications*, vol. 41, pp. 2128–2136, July 2021.

[17] W. Tuerxun, X. Chang, G. Hongyu, J. Zhijie, and Z. Huajian, "Fault diagnosis of wind turbines based on a support vector machine optimized by the sparrow search algorithm," *IEEE Access*, vol. 9, pp. 69307–69315, 2021.

[18] R. Wu, H. Huang, J. Wei, C. Ma, Y. Zhu, Y. Chen, and Q. Fan, "An improved sparrow search algorithm based on quantum computations and multi-strategy enhancement," *Expert Systems with Applications*, vol. 215, p. 119421, April 2023.

[19] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: sparrow search algorithm," *Systems Science & Control Engineering*, vol. 8, pp. 22–34, January 2020.

[20] Y. Xue, B. Xue, and M. Zhang, "Self-adaptive particle swarm optimization for large-scale feature selec-

tion in classification," *ACM Transactions on Knowledge Discovery from Data*, vol. 13, pp. 1–27, September 2019.

[21] S. Yan, P. Yang, D. Zhu, W. Zheng, and F. Wu, "Improved sparrow search algorithm based on iterative local search," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–31, December 2021.

[22] X.-S. Yang, S. Deb, and S. K. Mishra, "Multi-species cuckoo search algorithm for global optimization," *Cognitive Computation*, vol. 10, pp. 1085–1095, June 2018.

[23] J. Yuan, Z. Zhao, Y. Liu, B. He, L. Wang, B. Xie, and Y. Gao, "DMPPT control of photovoltaic microgrid based on improved sparrow search algorithm," *IEEE Access*, vol. 9, pp. 16623–16629, 2021.

[24] C. Zhang and S. Ding, "A stochastic configuration network based on chaotic sparrow search algorithm," *Knowledge-Based Systems*, vol. 220, p. 106924, May 2021.

[25] J. Zhang, K. Xia, Z. He, Z. Yin, and S. Wang, "Semisupervised ensemble classifier with improved sparrow search algorithm and its application in pulmonary nodule detection," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–18, February 2021.

[26] Y. Zhu and N. Yousefi, "Optimal parameter identification of PEMFC stacks using adaptive sparrow search algorithm," *International Journal of Hydrogen Energy*, vol. 46, pp. 9541–9552, February 2021.

Biography

Yehua Chen, with a master's degree and associate professor, works in Hebei University of Science and Technology and Engineering, and studies intelligent optimization and machine learning.

Junfu Xi, with a master's degree and an associate professor, works at Hebei University of Science and Technology and Engineering Currently, His main research interests include network security and intelligent optimization.

Xia Liu, master's degree, lecturer, working in Xingtai University, research field of intelligent control.

Huan Gao, with a master's degree and lecturer, professional Hebei University of Technology and Engineering, main research interests include AI, software engineering.

New Post-quantum Blockchain Privacy Protection Scheme Based on the Signcryption

Wang-Ke Yu and Xi-En Cheng
(Corresponding author: Wang-Ke Yu)

School of Information Engineering, Jingdezhen Ceramic University
Jingdezhen, China
Email: ywkyyy@163.com

(Received Oct. 20, 2022; Revised and Accepted Apr. 12, 2023; First Online Apr. 27, 2023)

Abstract

This paper proposes a blockchain signcryption scheme against quantum computing, which uses the signcryption to protect the blockchain based on the zero-knowledge argument of knowledge systems in the lattice to improve network security and have post-quantum computing. It can effectively protect the security of core data, such as the true identity of the blockchain, resist quantum computing, and hide the accurate identity information of blockchain users. Using a signcryption to verify the legitimacy of users and the validity of the signcryption can't obtain the accurate identity information of users in the whole process of signcryption, which well protects the privacy of blockchain.

Keywords: Blockchain; Privacy; Signature; Zero-knowledge Argument of Knowledge

1 Introduction

With the continuous increase of the volume of data and the continuous improvement of the intrinsic value of data, it becomes particularly important to make full use of the centralized data value in major traditional organizations and play its role as a factor of pro-duction in the era of the digital economy. However, as the carrier of data value, the traditional data trading platform has many problems, such as the loss of data ownership and the disclosure of original data information. How to make data transactions while ensuring the invisibility of the data and the determination of data ownership has become an urgent problem to be solved [17, 22, 28, 31]. Blockchains can store different contents on the chain through the same data format, to realize the secure sharing of content. At present, blockchain technology has received widespread attention, is being derived into a new form of industry, and has become a new momentum of economic development [1–3]. The application of blockchain has been extended to many fields, such as intelligent manufacturing, supply chain management, the internet of things, digital currency, health care, and so on [8–11, 16, 18, 29]. Due to the rapid development of quantum computer technology and the continuous improvement of the number of qubits of quantum computers, quantum computers can quickly solve difficult exponential problems, so when the number of qubits of quantum computers increases to a certain extent, it will bring serious security risks to the existing computer networks based on classical public key cryptography algorithms. In practical applications, most of the existing protocols are generally constructed using traditional cryptographic primitives, such as Diffie-Hellman or Elliptic Curve Diffie-Hellman. In the future quantum era of universal quantum computers, these algorithms will pose a great threat [15, 21, 26, 33]. With the development of quantum computing, post-quantum cryptography has attracted much more attention.

The last decade has witnessed important progress in the field of computing on lattice-based [4, 6, 20, 30]. As mentioned earlier the security proofs for all these multi-signatures are either incomplete or rely on a non-standard heuristic assumption. In 2021, Doss *et al.* [14] proposed a Memetic optimization with cryptographic encryption for secure medical data transmission in IoT-based distributed systems that can resist inside key exchange by using the lattice signature technology. The application of the key exchange and signature scheme also includes hierarchical electronic voting for multiple regions, digital copyright management, and much more. For practical lattice-based zero-knowledge argument of knowledge (ZKAoK) systems, there are two main approaches in current literature: Stern-type Protocol and Fiat-Shamir with Abort [5, 12, 13, 19]. In 2019, Bootle *et al.* proposed the algebraic techniques for short exact lattice-based zero-knowledge argument of knowledge systems [7]. In 2021, Lyubashevsky *et al.* proposed a shorter lattice-based zero-knowledge argument of knowledge systems via one-time commitments [25]. The research on the problem of voter privacy data disclosure in the application of blockchain and the identity privacy protection scheme of blockchain applied to different scenarios not only has a certain theoretical value, but also has important practical significance. Hence, it is meaningful to construct a blockchain post-quantum signcryption scheme based on the zero-knowledge argument of the knowledge system, which can pro-vide useful information privacy and unforgeability.

The rest of the paper is organized as follows: in Section 2, we introduce some basic concepts and algorithms of lattice schemes. In Section 3, we give our post-quantum blockchain privacy protection scheme. In Section 4, we analyze the correctness and security. In Section 5, finally, we summarize the post-quantum blockchain privacy protection scheme.

2 Lattice Problem

Definition 1. *Let Λ be an n-dimensional lattice and $\varepsilon > 0$. Then, the smoothing parameter $\eta_\varepsilon(\Lambda)$ is the smallest real $s > 0$ such that $\rho_{\frac{1}{\sqrt{2\pi}s}}(\Lambda^*\backslash\{0\}) \leq \varepsilon$.*

Lemma 1. *For any n-dimensional lattice Λ with basis B and $\varepsilon > 0$, we have:*

$$\eta_\varepsilon(\Lambda) \leq \left\|\tilde{B}\right\| \cdot \sqrt{\ln(2n/(1+1/\varepsilon))/\pi}.$$

Lemma 2. *Let Λ be an n-dimensional lattice. Then, for any $\varepsilon \in (0,1)$, $\sigma \geq \eta_\varepsilon(\Lambda)$ and and $c \in R^n$*

$$\rho_{\sigma,c}(\Lambda) \in \left[\frac{1-\varepsilon}{1+\varepsilon}, 1\right] \cdot \rho_\sigma(\Lambda).$$

Lemma 3. *Let $m, k > 1$, Λ be m-dimensional lattice and$c \in Z^m$. Then:*

$$\Pr_Z \leftarrow D_\sigma[|z| > k\sigma] \leq 2e^{\frac{-k^2}{2}}$$

$$\Pr_Z \leftarrow D_\sigma^m[\|z\|_2 > k\sigma\sqrt{m}] \leq k^m e^{\frac{m}{2}(1-k^2)}$$

$$\Pr_Z \leftarrow D_{\Lambda,\sigma,c}^m[\|z\|_2 > k\sigma\sqrt{m}] \leq 2k^m e^{\frac{m}{2}(1-k^2)}.$$

Lemma 4. *([27, 32]). Let $Q \in Z^{m\times n}$and Λ be an n-dimensional lattice. Then, for any $\sigma \in R_{>0}^m$ and $s \in R^m$ we have:*

$$\frac{\rho_\sigma(s)}{\sum_{z\in\Lambda}\rho_\sigma(Qz)} \leq \frac{\rho_\sigma(s)}{\sum_{z\in\Lambda}\rho_\sigma(s+Qz)} \leq \frac{1}{\sum_{z\in\Lambda}\rho_\sigma(Qz)}.$$

Theorem 1. *([32]). Let $A \in Z^{n\times m}$ and $W \in Z^{k\times m}$ be arbitrary matrices and denote $w_i \in Z^m$ to be the i-th row of W. Furthermore, suppose$\sigma = (\sigma_1, \sigma_2, \cdots, \sigma_k)$ satisfies for$\sigma_i \geq q^{n/m}\sqrt{\frac{ek}{m}}\,\|w_i\| + 2$. Then, for any $s \in R^k$, we have:*

$$\frac{\rho_\sigma(s)}{\sum_{z\in\Lambda_q^\perp(A)}\rho_\sigma(s+Wz)} \leq \frac{1}{2}.$$

Definition 2. *($\mathrm{MLWE}_{n,m,\chi}$) ([24, 32]). Given $A \leftarrow R_q^{n\times m}$, a secret vector $s \leftarrow \chi^m$ and error vector $e \leftarrow \chi^n$, the $\mathrm{Module-LWE}$ problem with parameters $n, m > 0$ and an error distribution χ over R asks the adversary ψ to distinguish between the following cases: $(A, As+e)$ for A. Then, ψ is said to have advantages in solving $\mathrm{MLWE}_{n,m,\chi}$ if*

$$|\Pr[b=1|A \leftarrow R_q^{n\times m}; s \leftarrow \chi^m; e \leftarrow \chi^n; b \leftarrow \psi(A, As+e)]$$

$$-\Pr[b=1|A \leftarrow R_q^{n\times m}; \mathrm{b} \leftarrow R_q^n; b \leftarrow \psi(A, \mathrm{b})]| \geq \varepsilon.$$

Definition 3. $\mathrm{Module-SIS}$*(.$\mathrm{MSIS}_{n,m,B}$..) ([24]). Given $A \leftarrow R_q^{n\times m}$, the $\mathrm{Module-SIS}$ problem with parameters $n, m > 0$ and $0 < B < q$asks to find $z \in R_q^m$ such that $Az = 0$ over R_q and $0 < \|z\| < B$. An algorithm ψ is said to have advantages in solving $\mathrm{MSIS}_{n,m,B}$ if:*

$$\Pr[0 < \|z\| < B \wedge Az = 0|A \leftarrow R_q^{n\times m}; z \leftarrow \psi(A)] \geq \varepsilon.$$

Lemma 5. *(Lattice Trapdoors [6]).* $\mathrm{TrapSamp}(1^n, 1^m, q)$ *that, given any integers $n \geq 1$, $q \geq 2$, and sufficiently large $m = \Omega(n\log q)$, outputs a matrix $A \in Z_q^{n\times m}$ and a trapdoor matrix $T \in Z^{n\times m}$ such that the distribution of A is negl(n)-close to uniform.*

Lemma 6. *([23]). and $v \in R^n$. Then, for all $t > 0$, it holds that:*

1) $P_x \sim D_{\Lambda,\sigma}[\|x\|_2 > \sigma\sqrt{n}] < 2^{-2n}$,

2) $P_x \sim D_{\Lambda,\sigma}[\|x\|_\infty > \sigma\log_2 n] \leq 2ne^{-\pi\log_2^2 n}$,

3) $P_x \sim D_{\Lambda,\sigma}[|\langle x\rangle| > \sigma t\|v\|_2] \leq 2e^{-\pi t^2}$.

Lemma 7. *([19]). Let n, p be positive integers. Let Λ be a lattice of rank n, and let $V = [-p, p]^n$. Let $T = p\sqrt{5n(1+\delta)/8}$, where*

$$\delta = \sqrt{\frac{32(\lambda+1)}{25n\log_2 e}}.$$

Define h the distribution obtained by sampling α from $[-p, p]$ and s from ψ_1^n and outputting $v = \alpha \cdot s$. Further, let $M > 1$, $t = \sqrt{(\lambda+2)/(\pi\log_2 e)}$ and definitely

$$\sigma_{\min} = \left(-t + \sqrt{\frac{t^2 + \ln(M)}{\pi}}\right)^{-1} \cdot T.$$

Let $\sigma \geq \sigma_{\min}$. We now define two distributions

P_1: *Sample $v \leftarrow h$ and $y \leftarrow D_{\Lambda,\sigma}$. Define $z = y + v$. Output (v, z) with probability*

$$\min\left(1, \frac{D_{\Lambda,\sigma}(z)}{M \cdot D_{\Lambda,\sigma}(z-v)}\right).$$

P_2: *Sample $v \leftarrow h$ and $z \leftarrow D_{\Lambda,\sigma}$. Output (v, z) with probability 1/M.*

Then, it holds that P_1 outputs something with probability at least $(1-2^{-\lambda})/M$, and that

$$\Delta(P_1, P_2) \leq 2^{-(\lambda+1)}(1+1/M) \leq 2^{-\lambda}.$$

3 Blockchain Privacy Protection Scheme

The blockchain privacy protection scheme involves a few parameters: a prime q_1 and prime q modulus, and integer dimensions $n, k, d, \ell, \ell_1, \ell_2, \tau, \lambda, M \geq 1$. More precisely, we define:

$$\ell = \ell_1 + \ell_2 + 2n$$

$$B_1 \leftarrow Z_q^{\ell_1 \times (\ell_2 + 2n)}$$

$$B_2 \leftarrow Z_q^{2n \times \ell_2}$$

$$B = \left[\begin{array}{cc} I_{\ell_1} & B_1 \\ 0 & I_{2n} B_2 \end{array} \right] \in Z_q^{(\ell_1 + 2n) \times \ell}.$$

We now present the zero-knowledge argument of knowledge ($ZKAoK$) protocol in [19]. First, let aCommit be an auxiliary commitment scheme with randomness space $\{0,1\}^n$ and message space $Z_q^{n+2\ell}$, and that is binding and hiding. The following interactive protocol involves a prover P with public input $A \in Z_q^{n \times m}$, $y \in Z_q^m$, and $M \in [n]^3$ with $|M| = n$ and private input $x \in Z_q^n$. The verifier V is only given the public input. In the protocol, the P must convince V in zero-knowledge that they know x verifying

$$\left\{ \begin{array}{l} y^{\mathrm{T}} = x^{\mathrm{T}} \cdot A \bmod q \\ \forall (h, i, j) \in M, x[h] = x[i]x[j] \bmod q \end{array} \right.$$

The Signcryption Generation: the Prover and the Verifier will implement the following algorithms to generate the signcryption:$KeyGen(A, \gamma^n)$, $P_i[A, B, M, H, \eta, \eta^{'}, y_j, x_i, t, r]$, $V_j[H, A, B, M, \eta, \eta^{'}, y_i, x_j, t]$. Assume that the Prover is i-th user and the Verifier is j-th user, the $KeyGen(A, \gamma^n)$ will be generated as shown in the following:

$KeyGen(A, \gamma^n)$:

1) Choose a random public matrix $A \in Z_q^{n \times m}$.

2) Choose random parameter x: $x_i \leftarrow \{-\tau, \cdots, \tau\}^n$, $i = 1, 2, \cdots, Q$, where i and Q represents the i-th user and total number of users respectively.

3) Compute: $y_i^{\mathrm{T}} = x_i^{\mathrm{T}} \cdot A \bmod q$, $i = 1, 2, \cdots, Q$.

The $P_i[A, B, M, H, \eta, \eta^{'}, y_j, x_i, t, r]$ will be generated as shown in the following:

Prover $P_i[A, B, M, H, \eta, \eta^{'}, y_j, x_i, t, r]$:

1) Choose random parameters: $\theta \leftarrow \{-\tau, \cdots, \tau\}^m$, $e \leftarrow \{-\tau, \cdots, \tau\}^n$, $e^{'} \leftarrow \{-\tau, \cdots, \tau\}$.

2) Compute: $\eta = q_1 \cdot A \cdot \theta + q_1 \cdot e \bmod q$.

3) Choose a random message μ: $\mu \in \{0,1\}^n$.

4) Choose a hash function H:

$$H : \{0,1\}^* \rightarrow \{v_1 : v_1 \in \{0,1\}^n\}$$

5) Compute: $\mu_H = H(\mu)$.

6) Compute: $\eta^{'} = q_1 \cdot y_j^{\mathrm{T}} \cdot \theta + q_1 \cdot e^{'} + \mu \bmod q$.

7) Sample Gaussian parameters: $r \leftarrow D^n_{Z_q, r}$, $s_1 \leftarrow D^{\ell}_{\sigma_1}$, $s_2 \leftarrow D^{\ell}_{\sigma_2}$.

8) Compute: $t^{\mathrm{T}} = r^{\mathrm{T}} A \bmod q$.

9) Let a and b be two n-dimension vectors. For $h \in [1, n]$, let (k_1, k_2, k_3) be the h-th element of M, compute:

$$a[h] = r[k_1] - r[k_2] \cdot x_i[k_3] - r[k_3] \cdot x_i[k_2]$$

$$b[h] = r[k_2] \cdot x_i[k_3]$$

$$c_1 = B \cdot s_1 + \left[\begin{array}{c} 0 \\ x_i \\ a \end{array} \right] \bmod q$$

$$c_2 = B \cdot s_2 + \left[\begin{array}{c} 0 \\ r \\ b \end{array} \right] \bmod q$$

10) Sample: $\rho \leftarrow \{0,1\}^k$.

11) Output the commit: $C_{aux} = \mathrm{aCommit}(t||c_1||c_2; \rho)$.

Prover sends the commit C_{aux} to the Verifier.

Verifier sample a parameter α:$\alpha \leftarrow [-p, p]$ to the Prover.

Prover $P_i[A, B, M, H, \eta, \eta^{'}, y_j, x_i, t, r]$:

1) Compute: $z_0 = \alpha \cdot x + r$.

2) Compute: $z_1 = \alpha \cdot s_1 + s_2$

3) Abort with probability $1 - \mathrm{p}(\alpha \cdot s_1, z_1)$, $\mathrm{p}(\alpha \cdot s_1, z_1) = \min(\frac{1, D_{\sigma_2}(z_1)}{M \cdot (z_1 - \alpha \cdot s_1)})$.

Prover sent the commit $(\mu_H, H, t, \eta, \eta^{'}, s_1, s_2, z_0, z_1)$ to the Verifier.

The $V_j[H, A, B, M, \eta, \eta^{'}, y_i, x_j, t]$ will be generated as shown in the following:

Verifier $V_j[H, A, B, M, \eta, \eta^{'}, y_i, x_j, t]$:

Let a and b be two n-dimension vectors. For $h \in [1, n]$, let (k_1, k_2, k_3) be the h-th element of M, compute:

$d[h] = \alpha \cdot z_0[k_1] - z_0[k_2] \cdot z_0[k_3]$ $\mu^{'} = ((\eta^{'} - x_j^{\mathrm{T}} \cdot \eta) \bmod q) \bmod q_1$

Accept if:

1) $C_{aux} = \mathrm{aCommit}(t||c_1||c_2; \rho)$.

2) $\|z_1\|_\infty \leq s_2 \log_2 \ell$.

3) $\|z_1\|_2 \leq s_2 \sqrt{\ell}$.

4) $z_0^{\mathrm{T}} \cdot A = \alpha \cdot y_i^{\mathrm{T}} + t^{\mathrm{T}}$.

5) $B \cdot z_1 + \left[\begin{array}{c} 0 \\ z_0 \\ d \end{array} \right] = \alpha \cdot c_1 + c_2 \bmod q$.

6) $\mu^{'}_H = H(\mu^{'}) = \mu_H$

4 Analysis

4.1 correctness

The correctness of the decryption in the post-quantum blockchain privacy protection scheme follows from our choice of parameters. Specifically, to show correctness, we follow the proof strategy, we first compute $\mu^{'} = ((\eta^{'} - x_j^{\mathrm{T}} \cdot \eta) \bmod q) \bmod q_1$. We have:

$$\mu^{'} = ((\eta^{'} - x_j^{\mathrm{T}} \cdot \eta) \bmod q) \bmod q_1$$

$$= ((q_1 \cdot y_j^{\mathrm{T}} \cdot \theta + q_1 \cdot e^{'} + \mu - x_j^{\mathrm{T}} \cdot (q_1 \cdot A \cdot \theta + q_1 \cdot e)) \bmod q) \bmod q_1$$

$$= ((q_1 \cdot y_j^{\mathrm{T}} \cdot \theta - q_1 \cdot x_j^{\mathrm{T}} \cdot A \cdot \theta + q_1 (e^{'} + x_j^{\mathrm{T}} \cdot e) + \mu) \bmod q) \bmod q_1$$

$$= ((q_1 \cdot (e^{'} + x_j^{\mathrm{T}} \cdot e) + \mu) \bmod q) \bmod q_1$$

Since we assumed $(n \cdot d \cdot \gamma + 1) \leq \frac{q}{2q_1} - \frac{1}{2}$ and $\|\mu\|_\infty \leq q_1/2$, then

$$\left\| q_1 \cdot (e^{'} + x_j^{\mathrm{T}} \cdot e) + \mu \right\|_\infty \leq q_1/2_H$$

Therefore there is no reduction modulo q_1 in $q_1 \cdot (e^{'} + x_j^{\mathrm{T}} \cdot e) + \mu$ and hence

$$\mu^{'} = ((q_1 \cdot (e^{'} + x_j^{\mathrm{T}} \cdot e) + \mu) \bmod q) \bmod q_1 = \mu$$

Then

$$\mu_H^{'} = H(\mu^{'}) = H(\mu) = \mu_H$$

4.2 Proof of $ZKAoK$

Proof. For the zero-knowledge argument of knowledge proof of the post-quantum blockchain privacy protection scheme, we follow the proof strategy from [32].

Completeness: by Lemma 7, it holds that the prover responds with probability at least $(1-2^{-\lambda})/M$, and that z1 is within statistical distance $2^{-(\lambda+1)}(1+1/M)$ of D_{σ_2}. We further condition on a non-aborting transcript. Lemma 6 combined with the union bound gives

$$P\left[\|z_1\|_\infty \geq s_2 \log_2 \ell \vee \|z_1\|_2 \geq s_2\sqrt{\ell}\right]$$
$$\leq 2^{-(\lambda+1)}(1+1/M) + 2^{-2\ell} + 2\ell e^{-\pi \log_2^2 \ell}$$

The equation $z_0^{\mathrm{T}} \cdot A = \alpha \cdot y_i^{\mathrm{T}} + t^{\mathrm{T}}$ is easily verified as:

$$z_0^{\mathrm{T}} \cdot A = (\alpha \cdot x_i^{\mathrm{T}} + r^{\mathrm{T}}) \cdot A$$
$$= \alpha \cdot x_i^{\mathrm{T}} \cdot A + r^{\mathrm{T}} \cdot A$$
$$= \alpha \cdot y_i^{\mathrm{T}} + t^{\mathrm{T}}.$$

Let a and b be two n-dimension vectors. For $h \in [1, n]$, let (k_1, k_2, k_3) be the h-th element of M, we have:

$$d[h] = \alpha \cdot z_0[k_1] - z_0[k_2] \cdot z_0[k_3]$$

$$= \alpha \cdot (\alpha \cdot x_i[k_1] + r[k_1]) - (\alpha \cdot x_i[k_2] + r[k_2]) \cdot (\alpha \cdot x_i[k_3] + r[k_3])$$

$$= \alpha^2 \cdot (x_i[k_1] - x_i[k_2] \cdot x_i[k_3]) - r[k_2] \cdot r[k_3]$$
$$+ \alpha \cdot (r[k_1] - r[k_2] \cdot x_i[k_3] - r[k_3] \cdot x_i[k_2])$$

$$= \alpha \cdot a[h] + b[h] \bmod q$$

As a result, it holds that $d = \alpha \cdot a + b \bmod q$. It thus yields

$$B \cdot z_1 + \begin{bmatrix} 0 \\ z_0 \\ d \end{bmatrix} = B \cdot (\alpha \cdot s_1 + s_2) + \begin{bmatrix} o \\ \alpha \cdot x_i + r \\ \alpha \cdot a + b \end{bmatrix} \bmod q$$

$$= \alpha \cdot \left(B \cdot s_1 + \begin{bmatrix} 0 \\ x_i \\ a \end{bmatrix} \right) + \left(B \cdot s_2 + \begin{bmatrix} 0 \\ r \\ b \end{bmatrix} \right) \bmod q$$

$$= \alpha \cdot c_1 + c_2 \bmod q$$

In summary, the verifier will accept with all but negligible probability, thus, the protocol is complete with a completeness error of 1-1/M. □

Proof. Proof of Knowledge: Let (A, y, M) be a statement, let B be the public parameter. Suppose a cheating prover $\hat{P}$ can convince the verifier that he possesses a valid witness for (A, y, M) with probability $1/M + \varepsilon$ for some non-negligible ε, then we construct a knowledge extractor that can extract a valid witness for (A, y, M) via invoking $\hat{P}$. By the binding property of the auxiliary commitment, the extractor is able to obtain

$$(\alpha, t, c_1, c_2, z_0, z_1)$$

$$(\alpha^{'}, t, c_1, c_2, z_0^{'}, z_1^{'})$$

$$(\alpha^{''}, t, c_1, c_2, z_0^{''}, z_1^{''})$$

For distinct α, $\alpha^{'}$ and $\alpha^{''}$ that satisfies

$$\begin{cases} \|z_1\|_\infty \leq s_2 \log_2 \ell \\ \|z_1\|_2 \leq s_2\sqrt{\ell} \\ z_0^{\mathrm{T}} \cdot A = \alpha \cdot y_i^{\mathrm{T}} + t^{\mathrm{T}} \\ B \cdot z_1 + \begin{bmatrix} 0 \\ z_0 \\ d \end{bmatrix} = \alpha \cdot c_1 + c_2 \bmod q \end{cases}$$

$$\begin{cases} \left\|z_1^{'}\right\|_\infty \leq s_2 \log_2 \ell \\ \left\|z_1^{'}\right\|_2 \leq s_2\sqrt{\ell} \\ z_0^{'\mathrm{T}} \cdot A = \alpha^{'} \cdot y_i^{\mathrm{T}} + t^{\mathrm{T}} \\ B \cdot z_1^{'} + \begin{bmatrix} 0 \\ z_0^{'} \\ d \end{bmatrix} = \alpha^{'} \cdot c_1 + c_2 \bmod q \end{cases}$$

$$\begin{cases} \left\|z_1^{''}\right\|_\infty \leq s_2 \log_2 \ell \\ \left\|z_1^{''}\right\|_2 \leq s_2\sqrt{\ell} \\ z_0^{''\mathrm{T}} \cdot A = \alpha^{''} \cdot y_i^{\mathrm{T}} + t^{\mathrm{T}} \\ B \cdot z_1^{''} + \begin{bmatrix} 0 \\ z_0^{''} \\ d \end{bmatrix} = \alpha^{''} \cdot c_1 + c_2 \bmod q \end{cases}$$

For $h \in [1, n]$, let (k_1, k_2, k_3) be the h-th element of M, we have:

$$d[h] = \alpha \cdot z_0[k_1] - z_0[k_2] \cdot z_0[k_3]$$

$$d^{'}[h] = \alpha^{'} \cdot z_0^{'}[k_1] - z_0^{'}[k_2] \cdot z_0^{'}[k_3]$$
$$d^{''}[h] = \alpha^{''} \cdot z_0^{''}[k_1] - z_0^{''}[k_2] \cdot z_0^{''}[k_3]$$

Now, let $\Delta_1 = \alpha^{'} - \alpha$ and $\Delta_2 = \alpha^{''} - \alpha$. The output of the extractor is the vector $\tilde{x} = \Delta_1^{-1} \cdot (z_0^{'} - z_0)$,

$Honest - VerifierZero - Knowledge$: The simulator first retrieves the challenge α from ξ via feeding it with a commitment of 0 under the auxiliary commitment scheme. Then it:

1) $C_{aux} = \text{aCommit}(t||c_1||c_2; \rho)$.

2) Sample: $z_0 \leftarrow Z_q^n$, $\alpha \leftarrow [-p, p]$ and $z_1 \leftarrow D_{\sigma_2}$.

3) Compute: $t^{\mathrm{T}} \leftarrow z_0^{\mathrm{T}} \cdot A - \alpha \cdot y_i^{\mathrm{T}}$.

4) Sample: $c_1 \leftarrow Z_q^{\ell_1 + 2n}$.

5) For $h \in [1, n]$, let (k_1, k_2, k_3) be the h-th element of M, Then, $d[h] = \alpha \cdot z_0[k_1] - z_0[k_2] \cdot z_0[k_3]$.

6) Compute: $c_2 = B \cdot z_1 + \begin{bmatrix} 0 \\ z_0 \\ d \end{bmatrix} - \alpha \cdot c_1 \bmod q$.

7) Sample: $\rho \leftarrow \{0, 1\}^k$.

8) Computes $C_{aux} = \text{aCommit}(t||c_1||c_2; \rho)$ and $C_{aux}^{'} = \text{aCommit}(0; \rho)$.

9) Finally, with probability 1/M, output $(C_{aux}, \alpha, t, \eta, \eta^{'}, c_1, c_2, \rho, z_0, z_1)$ and with probability 1-1/M, output $(C_{aux}^{'}, \alpha, \perp)$.

Next, we argue that the output of the simulator is computationally indistinguishable from verifier's view in an interaction with an honest prover with a valid witness x for (A, y, M). More detailed proof reference [32]. □

4.3 Efficiency Analysis

In this section, we mainly focus on the proof computational complexity, verification computational complexity. First, we make a comparison of proof computational complexity and verification computational complexity between our signcryption scheme base on the zero-knowledge proof and other related zero-knowledge proof schemes, Maller *et al.* Scheme [26], Bünz *et al.* Scheme [8] and Ben-Sasson *et al.* Scheme [5]. The specific results of computational complexity comparison of the schemes are shown in Table.1.

Table 1: Computational complexity of all schemes

Scheme	**Proof Complexity**	**Verification Complexity**
Maller *et al.* Scheme [26]	$O(nm \log n)$	$O(l + \log n)$
Bünz *et al.* Scheme [8]	$O(nm \log n)$	$O(n \log n)$
Ben-Sasson *et al.* Scheme [5]	$O(nmpoly \log n)$	$O(poly \log n)$
Our Scheme	$O(n(k + l) \log n)$	$O((k + l) \log n)$

As depicted in Table.1, we make a comparison of proof computational complexity and verification computational complexity between our signcryption scheme base on the zero-knowledge proof and Maller *et al.* scheme [26], Bünz *et al.* scheme [8] and Ben-Sasson *et al.* scheme [5]. The time complexity of proof computational is $O(nm \log n)$ in Maller *et al.* [26] zero-knowledge proof scheme and Bünz *et al.* [8]. The time complexity of proof computational is $O(nmpoly \log n)$ in Ben-Sasson *et al.* [5] zero-knowledge proof scheme. The time complexity of proof computational is $O(n(k+l) \log n)$ in our signcryption scheme base on the zero-knowledge proof. The time complexity of verification computational is $O(l + \log n)$ in Maller *et al.* [26] zero-knowledge proof scheme. The time complexity of verification computational is $O(n \log n)$ in Bünz *et al.* [8] zero-knowledge proof scheme. The time complexity of verification computational is $O(poly \log n)$ in Ben-Sasson *et al.* [5] zero-knowledge proof scheme. The time complexity of verification computational is $O((k + l) \log n)$ in our signcryption scheme. Moreover, our signcryption scheme base on the zero-knowledge proof, and Plonk can Ben-Sasson *et al.* Scheme [5] achieve a constant level in the time complexity of verification computational, but the time complexity of verification computational of Maller *et al.* [26] zero-knowledge proof scheme and Bünz *et al.* [8] will increase with the logarithmic or logarithmic square speed with the circuit size. By comparing the results, our proposed the signcryption scheme base on the zero-knowledge proof has certain advantages in computational complexity.

5 Conclusions

In this paper, a post-quantum blockchain privacy protection scheme is proposed, which uses the signcryption to protect the blockchain based on zero-knowledge argument of knowledge systems in the lattice, so as to improve the network security and have anti-quantum computing. It can effectively protect the security of core data, such as the true identity of blockchain, resist quantum computing, and hide the real identity information of blockchain users. The use of a signcryption to verify the legitimacy of users and the validity of the signcryption can't obtain the true identity information of users in the whole process of signcryption, which well protects the privacy of blockchain. Future work, we will continue to study the post-quantum blockchain privacy protection scheme based on zero-knowledge argument of knowledge systems in the lattice that support a more flexible zero-knowledge argument of knowledge systems. Additionally, the efficiency of the post-quantum blockchain privacy protection scheme can be further improved.

Acknowledgments

This article was supported by the Special Project for Scientific and Technological Cooperation of Jiangxi Province Grant Number [20212BDH80021].

References

[1] E. Akhtarkavan, B. Majidi, and A. Mandegari, "Secure Medical Image Communication Using Fragile Data Hiding Based on Discrete Wavelet Transform and A Lattice Vector Quantization," *IEEE Access*, vol. 11, no. 1, pp. 9701–9715, 2023.

[2] M. M. K. Al Nuaimi, K. Rishal, N. V. Oommen, and P. Sherimon, "Blockchain Implementation Framework for Tracing the Dairy Supply Chain," *Lecture Notes on Data Engineering and Communications Technologies*, vol. 142, no. 1, pp. 551–560, 2023.

[3] Beck and Roman, "Beyond Bitcoin: The Rise of Blockchain World," *Computer*, vol. 51, no. 2, pp. 54–58, 2018.

[4] R. Behnia, M. O. Ozmen, and A. A. Yavuz, "Lattice-Based Public Key Searchable Encryption from Experimental Perspectives," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, pp. 1269–1282, November 2020.

[5] E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev, "Scalable, transparent, and post-quantum secure computational integrity," *Cryptology ePrint Archive*, vol. 2018, pp. 46–128, 2018.

[6] F. Benhamouda, J. Camenisch, S. Krenn, V. Lyubashevsky, and G. Neven. "Better Zero-Knowledge Proofs for Lattice Encryption and Their Application to Group Signatures,". in *Advances in Cryptology – ASIACRYPT 2014* (P. Sarkar and T. Iwata, eds.), vol. 8874, pp. 551–572. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. Series Title: Lecture Notes in Computer Science.

[7] J. Bootle, V. Lyubashevsky, and G. Seiler. "Algebraic Techniques for Short(er) Exact Lattice-Based Zero-Knowledge Proofs,". in *Advances in Cryptology – CRYPTO 2019* (A. Boldyreva and D. Micciancio, eds.), vol. 11692, pp. 176–202. Springer International Publishing, Cham, 2019. Series Title: Lecture Notes in Computer Science.

[8] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell, "Bulletproofs: Short proofs for confidential transactions and more," in *2018 IEEE symposium on security and privacy (SP)*, pp. 315–334. IEEE, 2018.

[9] P. Y. Chang, M.S. Hwang, C.C. Yang, "A blockchain-based traceable certification system", in *Security with Intelligent Computing and Big-data Services*, pp. 363-369, 2018.

[10] L. Chen, Q. Fu, Y. Mu, L. Zeng, F. Rezaeibagha, M. S. Hwang, "Blockchain-based random auditor committee for integrity verification", *Future Generation Computer Systems*, vol. 131, pp. 183-193, 2022.

[11] Y. H. Chen, L. C. Huang, I. C. Lin, M. S. Hwang, "Research on the secure financial surveillance blockchain systems", *International Journal of Network Security*, vol. 22, no. 4, pp. 708-716, 2020.

[12] J.-P. D'anvers, M. Van Beirendonck, and I. Verbauwhede, "Revisiting Higher-Order Masked Comparison for Lattice-Based Cryptography: Algorithms and Bit-Sliced Implementations," *IEEE Transactions on Computers*, vol. 72, no. 2, pp. 321–332, 2023.

[13] D. Dharminder, C. B. Reddy, A. K. Das, Y. Park, and S. S. Jamal, "Post-Quantum Lattice-Based Secure Reconciliation Enabled Key Agreement Protocol for IoT," *IEEE Internet of Things Journal10*, vol. 10, no. 3, pp. 2680–2692, 2023.

[14] S. Doss, J. Paranthaman, S. Gopalakrishnan, A. Duraisamy, S. Pal, B. Duraisamy, C. Le Van, *, and D.-N. Le, "Memetic Optimization with Cryptographic Encryption for Secure Medical Data Transmission in IoT-based Distributed Systems," *Computers, Materials & Continua*, vol. 66, no. 2, pp. 1577–1594, 2021.

[15] V. Farzaliyev, J. Willemson, and J. K. Kaasik, "Improved lattice-based mix-nets for electronic voting," *IET Information Security*, vol. 17, no. 1, pp. 18–34, 2023.

[16] M. J. Gabbay, "Algebras of UTxO blockchains," *Mathematical Structures in Computer Science*, vol. 31, no. 9, pp. 1034–1089, 2021.

[17] N. Hamian, M. Bayat, M. R.Alaghband, Z. Hatefi, and S. M. Pournaghi, "Blockchain-based User Re-enrollment for Biometric Authentication Systems," *International Journal of Electronics and Information Engineering*, vol. 14, no. 1, pp. 18–38, 2022.

[18] N. Islam, Y. Marinakis, S. Olson, R. White, and S. Walsh, "Is BlockChain Mining Profitable in the Long Run," *IEEE Transactions on Engineering Management*, vol. 70, no. 2, pp. 386–399, 2023.

[19] R. L. A. J. Corentin and S. Olivier, "Lattice-based signature with efficient protocols, revisited," *eprint.iacr.org*, vol. eprint.iacr.org/2022/509, pp. 1–46, 2022.

[20] D. Li, H. Chen, C. Zhong, T. Li, and F. Wang, "A New Self-Certified Signature Scheme Based on NTRUSing for Smart Mobile Communications," *Wireless Personal Communications*, vol. 96, pp. 4263–4278, October 2017.

[21] H. Li, F. Guo, L. Wang, J. Wang, B. Wang, and C. Wu, "A Blockchain-Based Public Auditing Protocol with Self-Certified Public Keys for Cloud Data," *Security and Communication Networks*, vol. 2021, pp. 1–10, February 2021.

[22] L. Liu and J. Cao, "Analysis of One Lightweight Authentication and Key Agreement Scheme for Internet of Drones," *International Journal of Electronics and Information Engineering*, vol. 13, no. 4, pp. 142–148, 2021.

[23] V. Lyubashevsky. "Lattice Signatures without Trapdoors,". in *Advances in Cryptology – EUROCRYPT 2012* (D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell,

M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, D. Pointcheval, and T. Johansson, eds.), vol. 7237, pp. 738–755. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. Series Title: Lecture Notes in Computer Science.

[24] V. Lyubashevsky, N. K. Nguyen, and M. Plancon. "Efficient Lattice-Based Blind Signatures via Gaussian One-Time Signatures,". in *Public-Key Cryptography – PKC 2022* (G. Hanaoka, J. Shikata, and Y. Watanabe, eds.), vol. 13178, pp. 498–527. Springer International Publishing, Cham, 2022. Series Title: Lecture Notes in Computer Science.

[25] V. Lyubashevsky, N. K. Nguyen, and G. Seiler. "Shorter Lattice-Based Zero-Knowledge Proofs via One-Time Commitments,". in *Public-Key Cryptography – PKC 2021* (J. A. Garay, ed.), vol. 12710, pp. 215–241. Springer International Publishing, Cham, 2021. Series Title: Lecture Notes in Computer Science.

[26] M. Maller, S. Bowe, M. Kohlweiss, and S. Meiklejohn, "Sonic: Zero-knowledge snarks from linear-size universal and updatable structured reference strings," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, vol. 2019, pp. 2111–2128, 2019.

[27] D. Micciancio and O. Regev, "Worst-Case to Average-Case Reductions Based on Gaussian Measures," *SIAM Journal on Computing*, vol. 37, pp. 267–302, January 2007.

[28] M. M. Nabi and F. Nabi, "Cybersecurity Mechanism and User Authentication Security Methods," *International Journal of Electronics and Information Engineering*, vol. 14, no. 1, pp. 1–9, 2022.

[29] N. Tahat, A. K. Alomari, O. M. Al-Hazaimeh, and M. F. Al-Jamal, "An efficient self-certified multi-proxy signature scheme based on elliptic curve discrete logarithm problem," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 23, pp. 935–948, May 2020.

[30] K. Yamamura, Y. Wang, and E. Fujisaki, "Improved lattice enumeration algorithms by primal and dual reordering methods," *IET Information Security*, vol. 17, no. 1, pp. 35–45, 2023.

[31] L. Yan, X. Wang, and S. Yin, "Campus Garbage Image Classification Algorithm Based on New Attention Mechanism," *International Journal of Electronics and Information Engineering*, vol. 13, no. 4, pp. 131–141, 2021.

[32] R. Yang, M. H. Au, Z. Zhang, Q. Xu, Z. Yu, and W. Whyte. "Efficient Lattice-Based Zero-Knowledge Arguments with Standard Soundness: Construction and Applications,". in *Advances in Cryptology – CRYPTO 2019* (A. Boldyreva and D. Micciancio, eds.), vol. 11692, pp. 147–175. Springer International Publishing, Cham, 2019. Series Title: Lecture Notes in Computer Science.

[33] J. Zhang, J. Cui, H. Zhong, Z. Chen, and L. Liu, "PA-CRT: Chinese Remainder Theorem Based Conditional Privacy-Preserving Authentication Scheme in Vehicular Ad-Hoc Networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, pp. 722–735, March 2021.

Biography

Wang-Ke Yu received his Ph.D. in 2011 from the School of Computer Network and Security, Xidian University, China. His research interests include information security, cryptography and network security.

Xi-En Cheng entered Fudan University in 2011 to pursue a doctorate in computer vision, and received a doctorate in science in 2016. His research interests include information security, cryptography and network security.

Study on Protection of Genuine Digital Music Works by a Watermarking Algorithm

Juan Gao and Baike Zuo
(Corresponding author: Baike Zuo)

School of Music, Hebei Institute of Communications
Shijiazhuang 050073,China
Email: bk98493@163.com

(Received Apr. 9, 2022; Revised and Accepted Mar. 23, 2023; First Online Apr. 27, 2023)

Abstract

The protection of genuine digital music works is critical. In this paper, a watermarking algorithm was designed based on the technique of discrete wavelet transform and vector norms to achieve the protection of music works. The experiments on different types of digital music works showed that the watermarking algorithm designed in this paper had good security, a high mean opinion score and signal-to-noise ratio, and good imperceptibility, and it kept a high normalized coefficient and low bit error rate in the face of noise addition and resampling attacks and had an embedding capacity greater than 20 bps. The results demonstrate the effectiveness of the designed watermarking algorithm in protecting genuine digital music works.

Keywords: Digital Music; Genuine Protection; Robustness; Watermarking Algorithm

1 Introduction

With the continuous development of computer and Internet, traditional compact discs and tapes are gradually fading out of the market, music is getting closer and closer to digitalization, and the Internet is gradually becoming the way people enjoy music works, which, at the same time, brings the problem of rampant piracy [12]. In order to achieve protection of genuine digital music works, digital right management (DRM) has emerged [31], but the great inconvenience of DRM is not conducive to its practical promotion [8, 26]. Digital watermarking has a wide range of applications in digital media [6, 7, 13, 15] and plays a good role in the protection of genuine images and videos [23], and it can also be applied to audio signals.

Watermarking algorithms for audio signals has become a key issue for researchers to focus on [10]. Mosleh *et al.* [19] proposed a watermarking algorithm based on discrete cosine transformation (DCT) and LU decomposition and found through experiments that it had a good performance. Mohammed *et al.* [18] used discrete wavelet transform (DWT) to decompose the signal and then used DCT to encrypt the watermarked image. They found that the DCT method was not easy to be detected and the average signal-to-noise ratio (SNR) reached 61 dB.

Safitri *et al.* [22] proposed a watermark embedding method combining compressed sampling, DWT, and QR decomposition to embed the watermark into audio by quantization index modulation (QIM) and found through experiments that the SNR of the method was greater than 20 dB, indicating good robustness. Dronyuk *et al.* [9] designed a digital watermarking method based on generalized Fourier, Hartley transform, and Ateb function and verified the stability of the method in supporting the security of audio and image. This paper designed a watermarking algorithm based on DWT, verified the security and robustness of the algorithm through experimental analysis, and proved the reliability of the algorithm for genuine digital music protection. The designed watermarking algorithm can be applied in actual music works. This work provides a new method for the protection of audio signals.

2 Protection of Genuine Digital Music Works

DRM realizes the protection of genuine works through controlling the right to use the works. The technology strictly manages the playing and copying of works but brings greater inconvenience to the actual appreciation of music works [21]. Music works protected by DRM have a strong exclusivity, which restricts the use environment of the player and affects the experience of enjoying the works, which is not conducive to the promotion of the technology.

Digital watermarking is a method to protect works by embedding watermark information without changing the content of the original work [1]. Since the perception of the human ear is more sensitive than that of the human eye [20], modifications to the audio signal can be easily perceived by the human ear, so an excellent watermarking algorithm needs not only to have good resistance to attacks but also to be able to guarantee the quality of the musical work. The evaluation of watermarking algorithms includes the following three main aspects.

The subjective evaluation of perceptibility is based on the mean opinion score (MOS) [4], which refers to the listener's subjective perception of the music piece after embedding the watermark. The evaluation criteria of MOS are shown in Table 1.

Table 1: MOS evaluation criteria

Score	Audio indicators	Description
5	Excellent	Imperceptible
4	Good	Slightly perceptible
3	Medium	Perceptible, slightly unpleasant
2	Poor	Obviously perceptible, but tolerable
1	Very poor	Unbearable

The objective evaluation of perceptibility is based on the peak signal-to-noise ratio (PSNR) [27], which is a measure of the quality of the music piece after embedding the watermark. Let the audio signal before and after embedding the watermark be $x(n)$ and $x'(n)$ and the audio length be L. The PSNR is calculated by:

$$PSNR = 10\log_{10}(\frac{\max_{0\leq n\leq L}\{x^2(n)\}}{\sum_{n=1}^{L}[x'(n)-x(n)]^2}$$

The robustness is generally evaluated using the bit error rate (BER) [29] and the normalized coefficient (NC) [2]. The calculation formulas of BER and NC are:

$$\begin{aligned} BER &= \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} A(i,j)\oplus B(i,j)}{M\times N} \\ NC &= \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} A(i,j)B(i,j)}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N} A^2(i,j)}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N} B^2(i,j)}} \end{aligned}$$

where A is the original watermark, B is the extracted watermark, $\oplus$ is the exclusive or operation, and M and N are the row and column of the watermark signal.

The unit of embedding capacity is bps. Let the watermark size be N and the length of audio signal be K. The embedding capacity is:

$$payload = \frac{N}{K}$$

According to the International Federation of the Phonographic Industry (IFPI) [16], this value needs to be greater than or equal to 20 bps.

3 Watermarking Algorithm Based on Vector Norms and Wavelet Transform

3.1 Watermark Pre-processing

A binary image is used as a watermark information for genuine digital music protection. It is preprocessed first in order to improve security. Arnold transform [24] is a method to achieve encryption by scrambling the coordinates of pixel points, defined as:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \bmod N,$$

where x and y are the pixel coordinates before scrambling, x' and y' are the pixel coordinates after scrambling, and N is the row or column width of the image.

After scrambling, the watermarked signal is reduced dimensionally. For a $N\times M$ watermarked image W, its one-dimensional binary sequence is $w'(x,y)$. The dimensionality reduction process is written as:

$$\begin{aligned} V &= \{v(k) = w'(x,y), \\ &\qquad 0 < i \leq N, 0 < j \leq M, k = i\times M + j\}. \end{aligned}$$

For a one-dimensional watermark, the encryption is performed using logistic mapping [28], and the relevant formula is:

$$x_(n+1) = \mu x_n(1-x_n),$$

where $x_n \in [0,1]$ and $\mu \in [0,4]$. When $3.569945 < \mu \leq 4$, the system is chaotic. Let $\mu = 3.8$ and initial value $x_0 = 0.6$, the system is in a chaotic state. Sequence x_n is obtained according to the above formula, and chaotic sequence μ_k is obtained after quantification. Then, it is processed by exclusive or along with $v(k)$ to obtain the watermark sequence after secondary encryption:

$$W'' = v(k)\oplus \mu_k, 1\leq i\leq N\times M.$$

3.2 Watermark Embedding

The principle of the watermarking algorithm designed in this paper is to firstly DWT the original audio signal, write down the low frequency coefficients in it as a vector, and then combine the vector parametrization to achieve the embedding of the watermark. The two theories involved in the process are as follows.

1) Vector norm [14]: For vector $A = (a_1, a_2, \cdots, a_n)$, its P-norm is ρ, defined as $\rho = A_P = (\sum_{i=1}^{n} |a_i|^P)^{1/P}$.

2) Wavelet transform: It is a method of signal processing [11], which solves the shortage of Fourier transform in dealing with abrupt signals [25] and can extract more useful information. It has good performance in signal processing [30], image processing [5], etc. Suppose there is square productable

function $\psi(t)$, $\psi(t) \in L^2(R)$, whose Fourier transform $\psi(\omega)$ satisfies $\int \frac{|\psi(\omega)|^2}{\omega} d\omega < \infty$, then the equation is called the admissible condition. Let the scale factor of $\psi(t)$ be α and the translation factor be τ, then $\psi(t)$ after translational expansion is written as: $\psi_{\alpha,\tau}(t) = \alpha^{-1/2}\psi(\frac{t-\tau}{\alpha}$, $a > 0$, $\tau \in R$, and $\psi_{\alpha,\tau}(t)$ is the non-interrupted wavelet basis function. Since most of the signals in practice are discretized digital signals, DWT is more commonly used [3].

α and τ are discretized: $\alpha = 2^j$, $\tau = 2^k T_s$, and then the discrete wavelet function is obtained: $\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}t - k)$. For $f(t)$, its discrete wavelet function is: $WT_f(j,k) = (f, \psi_{j,k}) = \int_R f(t)\psi_{j,\bar{k}}(t)dt$. After the signal is processed by DWT, the energy is mainly concentrated in the low-frequency component, so the watermark information can be embedded into it.

The process of watermark embedding based on vector norms and DWT is as follows.

1) The watermark is divided into $N \times M$ frames. Two-stage DWT processing is performed on the audio signal to get a low-frequency component cA_2 and two high-frequency components cD_1 and cD_2.

2) The low-frequency coefficient is denoted as vector D, which is divided into vectors $V1$ and $V2$: $V1 = \{D(i), 1 \leq i \leq L_c/2\}$, $V2 = \{D(i), (L_c/2) + 1 \leq i \leq L_c\}$, where L_c is the length of cA_2 and cD_2, $L_c = \frac{L}{N \times M \times 2^2}$ (L is the length of the audio signal).

3) The 2-norm of $V1$ and $V2$, i.e., $Norm_{V1}$ and $Norm_{V2}$, are calculated. Average value Norm is calculated. When the watermark bit to be embedded W'' is 1, the watermark is embedded according to the formula

$$\begin{aligned} Norm_{V1} &= Norm + q \\ Norm_{V2} &= Norm - q, \end{aligned}$$

when the watermark bit to be embedded W'' is 0, the watermark is embedded according to the formula

$$\begin{aligned} Norm_{V1} &= Norm - q \\ Norm_{V2} &= Norm + q, \end{aligned}$$

were q is the adjustable quantization strength, 0.03 here.

4) Vectors $V1'$ and $V2'$ are reconstructed using the revised norms, and the results are combined to obtain vector D'.

5) Inverse discrete wavelet transform (IDWT) is performed to obtain a frame of audio signal after embedding the watermark. The above steps are repeated until all watermark bits are embedded.

3.3 Watermark Extraction

The watermark extraction process is as follows.

1) The audio is divided into $N \times M$ frames for two-stage DWT processing. The low-frequency vector is denoted as D', which is divided into vectors $V1'$ and $V2'$.

2) The vector norms of $V1'$ and $V2'$, i.e., $Norm_{V1}$ and $Norm_{V2}$, are calculated. If $Norm_{V1} > Norm_{V2}$, then the watermark bit is 1; otherwise, it is 0.

3) Arnold transform and logistic mapping are used for decryption to get the watermark information.

4 Results and Analysis

Experiments were carried out in MATLAB2018 environment. Five different types of digital music (rock, pop, blues, classical, and jazz) were randomly selected from the network music library for experiments, and they were all wav format and monophonic. The sampling frequency was 44.1 kHz, 16 bit. A piece of 5 s was taken from every music as experimental audio signals. In the case of correct and incorrect keys, the extracted watermarked images are shown in Figures 1-3.

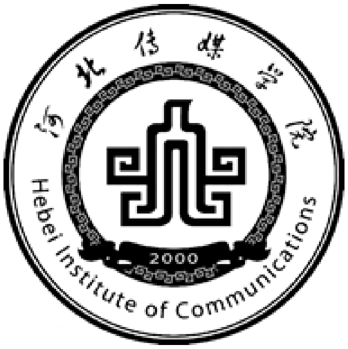


Figure 1: Original watermarked image

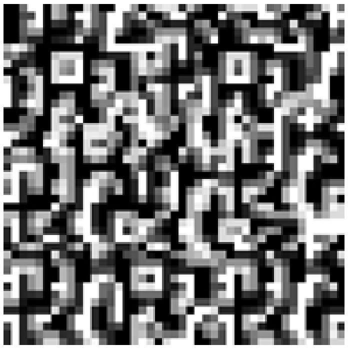


Figure 2: The extracted watermark image in the case of key error

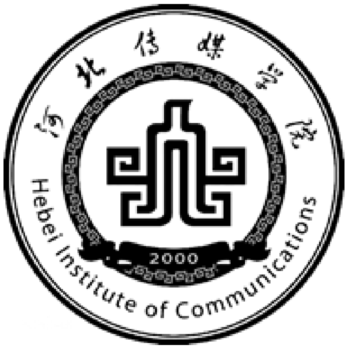


Figure 3: The extracted watermark image when the key is correct

It was seen from the comparison between Figures 1-3 that when the watermark was embedded using the watermarking algorithm designed in this paper, the correct watermarked image was not obtained in the case of incorrect key, while the complete watermarked image was extracted when the key was correct.

Ten students with normal hearing functions were selected as listeners to make MOS evaluation on the musics in a quiet environment, and their SNRs were calculated. The results are shown in Figure 4.

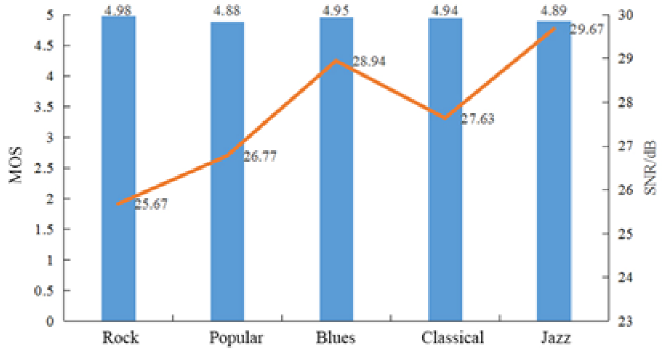


Figure 4: Imperceptibility evaluation results

It was seen from Figure 4 that, for these five types of music works, the MOS was always above 4.5 after embedding the watermark, with an average score of 4.93, indicating that the impact of embedding the watermark on the quality of music works was very small. Then, in terms of the objective perception, the SNR of different music genres was all above 20 dB after embedding the watermark, which was in line with the standard of greater than 20 dB as stipulated by IFPI.

The following attacks were performed on the watermarked audios.

1) Noise addition: Gaussian white noise with an expectation value of 0.01 and a variance of 0.05 was added.

2) Low-pass filter: a low-pass filter with six orders and a cut-off frequency of 10 kHz was used.

3) Requantization: the audio resolution was quantized from 16 bit to 8 bit and from 8 bit to 16 bit.

4) MP3 compression: the audios were converted from wav format to mp3 format and from map format to wav format.

5) Resampling: the audios were resampled using 22.05 kHz and then 44.1kHz.

6) Random cropping: ten locations were randomly selected to cut and remove 200 sampling points. The BER and NC of different types of music works under different attacks are listed in Table 2.

It was seen from Table 2 that the NC and BER of the audios was 1 and 0, respectively, under no attack. Different types of music works always maintained low BER (below 0.1) and high NC (above 0.9) under different attacks, indicating that the audios were less affected by attacks. The BER and NC values of different types of musical works were averaged, and the results are shown in Figure 5.

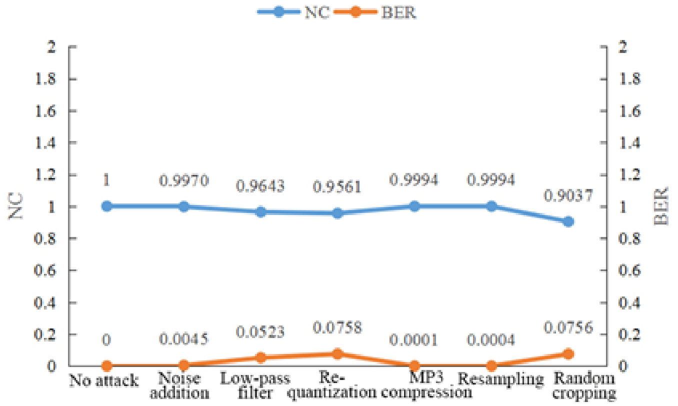


Figure 5: Results of robustness analysis of the watermarking algorithm

It was seen from Figure 5 that the BER of the audios was small under different attacks, the BER of the audios under random cropping was the largest, 0.0756, and the BER under MP3 compression was the smallest, 0.0001, which indicated that the watermarking algorithm designed in this paper maintained a low BER under different attacks. Then, in terms of NC, it was found that the NC of the audios under different attacks was very close to 1, with a maximum of 0.9994 and a minimum of 0.9037, both above 0.9, indicating that the designed watermarking algorithm maintained a high NC under different attacks. Finally, the embedding capacity was analyzed. The embedding capacity of the watermark was 87.64 bps after calculation, which met the IFPI standard–at least 20 bps.

Table 2: BER and NC of different music works

	NC/ BER	No attack	Noise addition	Low-pass filter	Requantization	MP3 compression	Resampling	Random cropping
Rock	NC	1	0.9964	0.9652	0.9568	0.9998	0.9995	0.9021
	BER	0	0.0044	0.0524	0.0763	0.0001	0.0004	0.0754
Popular	NC	1	0.9956	0.9646	0.9525	0.9998	0.9996	0.9056
	BER	0	0.0043	0.0526	0.0756	0.0001	0.0003	0.0764
Blues	NC	1	0.9974	0.9626	0.9556	0.9989	0.9989	0.9025
	BER	0	0.0051	0.0512	0.0758	0.0001	0.0004	0.0755
Classical	NC	1	0.9986	0.9646	0.9578	0.9997	0.9996	0.9056
	BER	0	0.0043	0.0525	0.0765	0.0001	0.0005	0.0752
Jazz	NC	1	0.9969	0.9646	0.9578	0.9989	0.9996	0.9025
	BER	0	0.0042	0.0526	0.0749	0.0001	0.0004	0.0754

5 Conclusion

This paper mainly studied the protection of genuine digital music works, designed a watermarking algorithm based on norms and DWT, and conducted experiments on actual music works. It was found that the designed watermarking algorithm had good security, high MOS, SNR above 20 dB, and good imperceptibility, and it maintained high NC and low BER even under different attacks, suggesting good robustness, and the embedding capacity also met the demand of digital music genuine protection. The designed watermarking algorithm can be promoted and applied in practice.

References

[1] Z. Al-Husseiny, "Using discrete wavelet transformation algorithm for authentication digital image watermark," *Journal of Engineering and Applied Sciences*, vol. 13, no. 11, pp. 4001-4008, 2018.

[2] M. H. Annaby, S. H. Basha, Y. M. Fouda, "Defect detection methods using boolean functions and the φ-coefficient between bit-plane slices," *Optics and Lasers in Engineering*, vol. 139, no. 6, pp. 106474, 2020.

[3] R. Arvanaghi, S. Danishvar, M. Danishvar, "Classification cardiac beats using arterial blood pressure signal based on discrete wavelet transform and deep convolutional neural network," *Biomedical Signal Processing and Control*, vol. 71, pp. 1-8, 2022.

[4] M. Barazzetta, M. Colombo, L. Bastianelli, F. Moglie, V. Mariani Primiani, R. Diamanti, D. Micheli, "Testing of VoLTE mean opinion score in reverberation chambers," *IET Science Measurement & Technology*, vol. 14, no. 8, pp. 949-954, 2020.

[5] H. K. Bhagya, N. Keshaveni, "Contrast enhancement technique using discrete wavelet transform with just noticeable difference model for 3D stereoscopic degraded video," *International Journal of Innovative Technology and Exploring Engineering*, vol. 10, no. 3, pp. 7-13, 2021.

[6] C. C. Chang, K. F. Hwang, M. S. Hwang, "A digital watermarking scheme using human visual effects", *Informatics*, vol. 24, no. 4, pp. 505–511, Dec. 2000.

[7] C. C. Chang, K. F. Hwang, M. S. Hwang, "A block based digital watermarks for copy protection of images," in *Asia-Pacific Conference on Communications*, vol. 2, pp. 977-980, 1999.

[8] Y. Chou, K. Anggriani, N. Wu, M. S. Hwang, "Research on e-book text copyright protection and anti-tampering technology", *International Journal of Network Security*, vol. 23, no. 5, pp. 739-749, 2021.

[9] I. Dronyuk, O. Fedevych, N. Kryvinska, "Constructing of digital watermark based on generalized fourier transform," *Electronics*, vol. 9, no. 7, pp. 1-13, 2020.

[10] R. R. Ginanjar, D. S. Kim, C. B. Moon, "High-capacity and transparent digital audio watermarking using rounding reduced-arc MPSK and a genetic algorithm," *IEIE Transactions on Smart Processing and Computing*, vol. 8, no. 1, pp. 49-57, 2019.

[11] M. Gogowski, S. Osowski, "Classical versus deep learning methods for anomaly detection in ECG using wavelet transformation," *Przeglad Elektrotechniczny*, vol. 97, no. 6, pp. 72-76, 2021.

[12] L. Guo, A. L. Wang, S. K. Li, J. P. Pang, K. Xin, S. X. Fan, F. P. Liu, "Authentication algorithm of secure digital halftone watermark based on SM4 algorithm," *IOP Conference Series: Materials Science and Engineering*, vol. 563, no. 5, pp. 1-5, 2019.

[13] M. S. Hwang, C. C. Chang, K. F. Hwang, "Digital watermarking of images using neural networks", *Journal of Electronic Imaging*, vol. 9, no. 4, pp. 548–555, Jan. 2000.

[14] T. Ke, L. Zhang, X. Ge, H. Lv, M. Li, "Construct a robust least squares support vector machine based on Lp-norm and L∞-norm," *Engineering Applications of Artificial Intelligence*, vol. 99, no. 3, pp. 104134, 2021.

[15] V. Korzhik, D. Flaksman, "Digital watermark system with an ability of its extraction from hard copies

of data," *Proceedings of Telecommunication Universities*, vol. 5, no. 3, pp. 75-85, 2019.

[16] Laing D. , "International federation of the phonographic industry

[17] . Moneyscience, 2009: 585-594.

[18] R. Mohammed, M. E. Abdulmunim, "Using Rubik's cube in fragile audio watermark encryption," *Diyala Journal of Engineering Sciences*, vol. 15, no. 3, pp. 103-124, 2020.

[19] M. Mosleh, S. Setayeshi, B. Barekatain, M. Mosleh, "High-capacity, transparent and robust audio watermarking based on synergy between DCT transform and LU decomposition using genetic algorithm," *Analog Integrated Circuits and Signal Processing*, vol. 100, no. 3, pp. 513-525, 2019.

[20] W. Mynarski, J. H. Mcdermott, "Ecological origins of perceptual grouping principles in the auditory system," *Proceedings of the National Academy of Sciences*, vol. 116, no. 50, pp. 25355-25364, 2019.

[21] S. Oliver, T. Winarta, "The lack of enforcement of the DRM policy," *Journal of Applied Information Communication and Technology*, vol. 7, no. 1, pp. 15-25, 2021.

[22] I. Safitri, G. Budiman, A. Putri, "Audio watermarking combined with compressive sampling based on QIM and DST-QR techniques," *Jurnal Elektronika dan Telekomunikasi*, vol. 19, no. 1, pp. 20, 2019.

[23] R. Sailaja, C. Rupa, A. Chakravarthy, "EPNN based high secure intensive hidden digital watermark application in telemedicine," *Advances in Modelling and Analysis*, vol. 56, no. 1, pp. 21-25, 2019.

[24] A. Shrivastava, J. Sharma, S. Purohit, "Image encryption based on fractional wavelet transform, arnold transform with double random phases in the HSV color domain," *Recent Advances in Computer Science and Communications*, vol. 15, no. 1, pp. 5-13, 2022.

[25] A. K. Singh, A. Saxena, N. Roy, U. Choudhury, "Inter-turn fault stability enrichment and diagnostic analysis of power system network using wavelet transformation-based sample data control and fuzzy logic controller," *Transactions of the Institute of Measurement and Control*, vol. 43, no. 12, pp. 2788-2798, 2021.

[26] C. Y. Tsai, C. Y. Yang, I. C. Lin, M. S. Hwang, "A survey of e-book digital right management", *International Journal of Network Security*, vol. 20, no. 5, pp. 998-1004, 2018.

[27] B. J. Tyler, R. Kassenböhmer, R. E. Peterson, D. T. Nguyen, M. Freitag, F. Glorius, B. J. Ravoo, "Denoising of mass spectrometry images via inverse maximum signal factors analysis," *Analytical Chemistry*, vol. 94, no. 6, pp. 2835-2843, 2022.

[28] J. Wang, K. Han, S. Fan, Y. Zhang, H. Tan, G. Joen, Y. Pang, J. Lin, "A logistic mapping-based encryption scheme for wireless body area networks," *Future Generation Computer Systems*, vol. 110, pp. 57-67, 2020.

[29] A. K. Yadav, P. K. Sahoo, Y. K. Prajapati, "Peak-to-average power ratio reduction and improved bit error rate performance of orthogonal frequency division multiplexing system using discrete Fourier transform precoder and root-based nonlinear companding," *Optical Engineering*, vol. 58, no. 7, pp. 076106.1-076106.9, 2019.

[30] L. Zamikhovsky, O. Zamikhovska, V. Pavlyk, "Research of the characteristics of acoustic processes using wavelet transformation for detecting a diagnostic sign of the technical state of gas pumping units," *Technology Audit and Production Reserves*, vol. 1, no. 2(57), pp. 32-36, 2021.

[31] X. Zhu, M. Cho, "The end of ownership?: An investigation of users' preferences and perceptions of ownership configurations," *Proceedings of the Association for Information Science & Technology*, vol. 55, no. 1, pp. 618-627, 2018.

Biography

Juan Gao, lecturer, master of arts, graduated from Guizhou University in 2010 and currently works at Hebei Institute of Communications. Her main research areas are music education and music aesthetics.

Baike Zuo, lecturer, master of education, graduated from Hebei University in 2017 and currently works at Hebei Institute of Communications. His main research areas are education management and music performance.

An Improved of Enhancements of a User Authentication Scheme

Min-Shiang Hwang[1], Hou-Wen Li[2], and Cheng-Ying Yang[3]

(Corresponding author: Cheng-Ying Yang)

Department of Computer Science, Information Engineering, Asia University, Taiwan[1]

The Ph.D. Program in Artificial Intelligence, Asia University, Taiwan[2]

Department of Computer Science, University of Taipei, Taiwan[3]

Email: cyang@utaipei.edu.tw

(Received Aug. 31, 2022; Revised and Accepted Apr. 23, 2023; First Online Apr. 30, 2023)

Abstract

With the rapid development of the Internet, telemedicine information systems (TLS) appear more and more around us. Nevertheless, the security of patient's medical information remains one of the most critical factors for the widespread adoption of TLS. Recently, Liu *et al.* proposed an improved three-factor authentication scheme. Through the BAN logic verification, their scheme is secure. However, the user needs to remember the long random number, which is impractical. In addition, their scheme could not withstand the fake smart card attack and a difficult-to-remember random number. In this article, we will improve their scheme for practicality and security.

Keywords: Password; Telemedicine Information System; Three-Factor Authentication; User Authentication

1 Introduction

User authentication schemes are designed to authenticate authorization services in servers over insecure channels. Users and servers can authenticate each other and then use the server's services using user authentication schemes [13,14,25]. Many scholars have proposed user authentication schemes [2,5,6,9,10,12,15,16,18–24,26–29]. A good user authentication scheme must meet security requirements and be simple and practical [3,8,11].

In 2013, Chang and Lee proposed a bright card-based user authentication scheme. Their scheme is easy to implement and practical [4]. However, Chang-Lee's scheme couldn't withstand online guessing identity, password, and denial of service attacks, as shown by Chiou *et al.*. Therefore, Chiou *et al.* also propose an improved user authentication scheme to withstand the vulnerability of Chang-Lee's scheme [7].

In 2015, Amin *et al.* proposed an RSA-based user authentication and key agreement scheme for telecare medical information systems [1]. They claimed their scheme provides good security protection against relevant security attacks. However, in 2019, Liu *et al.* showed that Amin *et al.*'s scheme could not be against the privileged insider attack, replay attack, stolen smart card attack, and user impersonation attack [17]. They thus proposed enhancements to Amin *et al.*'s user authentication scheme. This article will show that Liu *et al.*'s user authentication scheme could not withstand the fake smart card attack and a difficult-to-remember random number.

The rest of the paper is organized as follows. First, the review and weaknesses of Liu *et al.*'s scheme are described in Sections 2 and 3. Then, in Section 4, we propose an improved user authentication scheme that can resist all possible attacks mentioned in Section 3. Finally, Section 5 concludes the paper.

2 Review of Liu *et al.* User Authentication Scheme

In 2021, Liu *et al.* proposed a secure user authentication scheme [17]. We will review their scheme in this section. There are two primary entities in the Liu *et al.* scheme: the user U_i and server S. Furthermore, there are three phases: Registration, login and authentication, and password change.

2.1 Registration Phase

The procedures of the registration phase of Liu *et al.*'s scheme are listed as follows:

Step R1: The User U_i selects his/her identity (ID_i), password (PW_i), and a random number r.

Step R2: The User U_i extracts his/her fingerprint (T_i) as the second-factor authentication.

Step R3: The User U_i computes a strength password by $A_i = h(PW_i||r)$; a hiding fingerprint $F_i = H(T_i)$; an anonymous $RID = h(ID_i||r)$. Here, the sybmol $||$ denotes a concatenation operation.

Step R4: The User U_i sends $< RID_i, A_i, F_i >$ to S by a secure channel.

Step R5: Server S calculates W, B_i, and CID_i in the following equations:

$$\begin{aligned} W &= h(ID_s||x||RID_i) \\ B_i &= h(RID_i||A_i) \oplus W \\ CID_i &= Ex(RID_i||R_s). \end{aligned}$$

Here, ID_s is the server identity; x denotes a secure key of the Server S; E is an enciphering algorithm; and R_s denotes a random number by generating by S. Next, Server S stores these parameters $< A_i, B_i, CID_i, F_i, h(\cdot) >$ in smart card SD.

Step R6: The Server S sends the smart card SC to the User U_i by a secure channel.

2.2 Login & Authentication Phases

Identify applicable funding agency here. If none, delete this text box. The procedures of the login and authentication phases of Liu *et al.*'s scheme are listed as follows:

Step LA1: U_i starts SC. U_i keyins messages $\{ID_i, PW_i, r\}$, and extracts his/her fingerprint T_i. Next, U_i verifies the following parameters whether hold:

$$\begin{aligned} F_i^* &= h(T_i) \\ &\overset{?}{=} F_i; \\ A_i^* &= h(PW_i||r) \\ &\overset{?}{=} A_i; \\ RID_i^* &= h(ID_i||r) \\ &\overset{?}{=} RID_i. \end{aligned}$$

If the above equations are valid, the user generates a random number r_i and computes

$$\begin{aligned} W &= B_i \oplus h(RID_i||A_i); \\ C_1 &= r_iP; \\ C_2 &= r_i \oplus W; \\ C_4 &= h(RID_i||r_i||W||T_1). \end{aligned}$$

Here, r_i is a random number; P denotes a point on the elliptic curve; T_i denotes the current time of the smart card. Next, U_i sends messages $\{C_2, CID_i, C_4, T_1\}$ to server S.

Step LA2: Server checks $|T_s - T_1| < \Delta T$, where T_s is the current time of the server, and T is a threshold for delaying the transmission time. Next, S extracts RID_i from CID_i and computes

$$\begin{aligned} RID||R &= D_x(CID_i); \\ W &= h(ID_s||x||RID_i); \\ r_i^* &= C_2 \oplus W; \\ C_i^* &= r_i^*P; \\ C_4^* &= h(RID_i||r_i^*||W||T_1). \end{aligned}$$

Here D denotes a deciphering algorithm. S verifies whether $C_4^* = C_4$ trues; If it is true, the server produces a random number rj and computes

$$\begin{aligned} SK &= r_jC_1^*; \\ D_1 &= r_jP; \\ L_i &= h(RID_i^*||h(D_1)||W||T_2); \\ G_1 &= D_1 + C_1^*; \\ CID_i' &= Ex(RID_i||R'). \end{aligned}$$

Here r_j denotes a random number; T_2 is the current time of S. Next, S submits messages $\{CID_i', G_1, L_i, T_2\}$ to the user.

Step LA3: U_i verifies $|T_c - T_2| < \Delta T$, here T_c is the current time of SC. Next, SC computes

$$\begin{aligned} D_1^* &= G_1 - C_1^*; \\ L_i^* &= h(RID_i||h(D_1^*)||W||T_2); \\ SK &= r_iD_1^*; \\ &= r_ir_jP. \end{aligned}$$

Here, SK is the session sharing by the user U_i and server S. SC verifies whether $L_i^* = L_i$ trues, if it is true, calculates and sends $Z_i = h(RID_i||SK)$ to server.

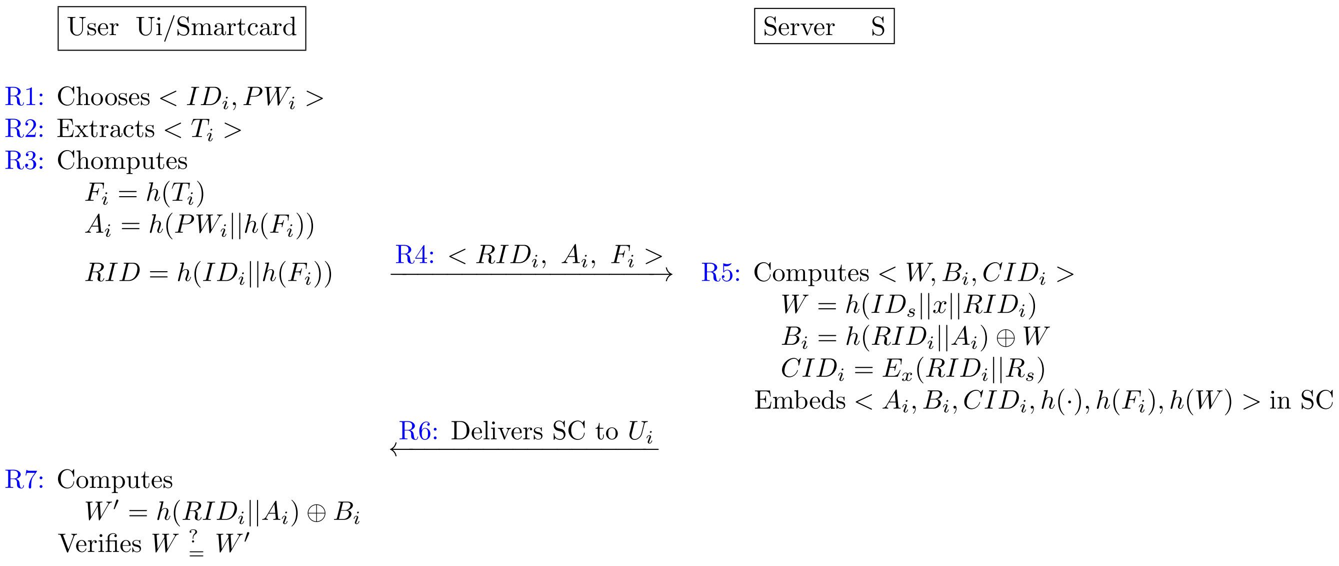
Step LA4: S computes $Z_i^* = h(RID_i||SK)$ and verifies whether $Z_i^* = Z_i$ holds.

2.3 Password Change Phase

Identify applicable funding agency here. If none, delete this text box. The procedures of the password change phase of Liu *et al.*'s scheme is listed as follows:

Step PC1: U_i starts smart card SC. U_i keys in messages $\{ID_i, PW_i, r\}$, and extracts his/her fingerprint T_i. Next, U_i verifies the following parameters whether hold:

$$\begin{aligned} F_i^* &= h(T_i) \\ &\overset{?}{=} F_i; \\ A_i^* &= h(PW_i||r) \\ &\overset{?}{=} A_i; \\ RID_i^* &= h(ID_i||r) \\ &\overset{?}{=} RID_i. \end{aligned}$$



User Ui/Smartcard

Server S

R1: Chooses $< ID_i, PW_i >$

R2: Extracts $< T_i >$

R3: Chomputes

$F_i = h(T_i)$

$A_i = h(PW_i || h(F_i))$

$RID = h(ID_i || h(F_i))$

$\xrightarrow{\text{R4: } < RID_i,\ A_i,\ F_i >}$

R5: Computes $< W, B_i, CID_i >$

$W = h(ID_s || x || RID_i)$

$B_i = h(RID_i || A_i) \oplus W$

$CID_i = E_x(RID_i || R_s)$

Embeds $< A_i, B_i, CID_i, h(\cdot), h(F_i), h(W) >$ in SC

$\xleftarrow{\text{R6: Delivers SC to } U_i}$

R7: Computes

$W' = h(RID_i || A_i) \oplus B_i$

Verifies $W \overset{?}{=} W'$

Figure 1: The registration phase of the proposed scheme

If the above equations are valid, U_i produces a new password $PW_{i_{new}}$ and computes

$$\begin{aligned} A_{i_{new}} &= h(PW_{i_{new}} || r); \\ B_{i_{new}} &= h(RID_i || A_{i_{new}}) \oplus W. \end{aligned}$$

Replaces $< A_i, B_i >$ with $< A_{i_{new}}, B_{i_{new}} >$ in Smart Card SC.

3 Weakness of Liu *et al.* User Authentication Scheme

This section shows that Liu *et al.*'s user authentication scheme [17] could not withstand the fake smart card attack and a difficult-to-remember random number.

3.1 The Fake Smart Card Attack

Step R6‘: The Hacker sends the fake smart card SC to the User U_i. The fake smart card has installed malicious software in order to steal the user's U_i fingerprint information.

Step LA1‘: U_i inserts the fake SC. U_i keys in messages $\{ID_i, PW_i, r\}$, and extracts his/her fingerprint T_i. Once the fake smart card obtains the fingerprint information, it sends the fingerprint information to the hacker and displays a message, "the smart card failed and should replace it", with the user U_i. Since the hacker has obtained the fingerprint information of the user U_i, the hacker can forge the identity of the user U_i and register with the Server.

3.2 A Difficult-To-Remember Random Number

Step LA1‘: U_i starts SC. U_i keys in messages $\{ID_i, PW_i, r\}$, and extracts his/her fingerprint T_i. In this step, the user U_i needs to know the random number r, at least 32 bits. Generally, It is difficult for users to remember a 32-bit random number.

4 The Proposed User Authentication Scheme

In the proposed scheme, there are also three phases: Registration, login and authentication, and password change.

4.1 Registration Phase

The registration phase of the proposed scheme is shown in Figure 1. The procedures of the registration phase of the proposed scheme are listed as follows:

Step R1: The User U_i selects his/her identity (ID_i), password (PW_i).

Step R2: The User U_i extracts his/her fingerprint (T_i) as the second-factor authentication.

Step R3: The User U_i computes a hiding fingerprint $F_i = h(T_i)$; a strength password by $A_i = h(PW_i || h(F_i))$; an anonymous $RID = h(ID_i || h(F_i))$.

Step R4: The User U_i sends $< RID_i, A_i, F_i >$ to S by a secure channel.

User Ui/Smartcard

Server S

LA1: Insert the smart card SC

Keys $< ID_i, PW_i >$ and Extracts T_i

Verifies

$$F_i^* = H(T_i) \overset{?}{=} F_i$$
$$A_i^* = h(PW_i||h(F_i^*)) \overset{?}{=} A_i$$
$$RID_i^* = h(ID_i||h(F_i^*)) \overset{?}{=} RID_i$$

If the above equations are valid,

U_i generates r_i and computes

$$W = B_i \oplus h(RID_i||A_i)$$
$$C_1 = r_i \cdot P$$
$$C_2 = r_i \oplus W$$
$$C_4 = h(RID_i||r_i||W||T_1)$$

$$\xrightarrow{\{C_2, CID_i, C_4, T_i, \}\text{ to S}}$$

LA2: Checks $|T_s - T_1| \overset{?}{<} \Delta T$

Extrocts RID_i from CID_i

Computes

$$RID||R = D_x(CID_i)$$
$$W = h(ID_s||x||RID_i)$$
$$r_i^* = C_2 \oplus W$$
$$C_1^* = r_i^* \cdot P$$
$$C_4^* = h(RID_i||r_i^*||W||T_1)$$

Checks $C_4^* \overset{?}{=} C_4$

Generates random number r_j

Computes

$$D_1 = r_j \cdot P$$
$$SK = r_j \cdot C_1^*$$
$$G_1 = D_1 + C_1^*$$
$$L_i = h(RID_i^*||h_1(D_1)||W||T_2)$$
$$CID_i' = E_x(RID_i||R_s')$$

$$\xleftarrow{\{CID_i', G_1, L_i, T_2\}}$$

LA3: Verifies $|T_c - T_2| \overset{?}{<} \Delta T$

Computes

$$D_i^* = G_1 - C_1^*$$
$$L_i^* = h(RID_i||h(D_1^*)||W||T_2)$$
$$SK = r_i \cdot D_1^* = r_i \cdot r_j \cdot P$$

Checks $L_1^* \overset{?}{=} L_i$

Computes $Z_i = h(RID_i||SK)$

$$\xrightarrow{\{Z_i\}}$$

LA4: Computes

$$Z_i^* = h(RID_i||SK)$$

Checks $Z_i \overset{?}{=} Z_i$

Figure 2: The login and authentication phases of the proposed scheme

Step R5: S calculates W, B_i, and CID_i as follows:

$$\begin{aligned} W &= h(ID_s||x||RID_i) \\ B_i &= h(RID_i||A_i) \oplus W \\ CID_i &= Ex(RID_i||Rs). \end{aligned}$$

Here, ID_s is the server identity; x denotes a secure key of the Server S; E is an enciphering algorithm; and R_s denotes a random number by generating by S. Next, Server S stores these parameters $< A_i, B_i, CID_i, h(\cdot), h(F_i), h(W) >$ in SD.

Step R6: The Server S sends the smart card SC to the User U_i by a secure channel.

Step R7: The User U_i takes A_i and B_i and computes

$$W‘ = h(RID_i||A_i) \oplus B_i.$$

Next, U_i verifies the W whether equal or to $W‘$. If it holds, the smart card is issued by Server S.

4.2 Login and Authentication Phases

The login & authentication phases of the proposed scheme are shown in Figure 2. The procedures of the login and authentication phases of the proposed scheme are listed in the following:

Step LA1: U_i starts SC. U_i keys in messages $\{ID_i, PW_i\}$, and extracts his/her fingerprint T_i. Next, U_i verifies the following parameters whether hold:

$$\begin{aligned} F_i^* &= h(T_i) \\ &\overset{?}{=} F_i; \\ A_i^* &= h(PW_i||h(F_i^*)) \\ &\overset{?}{=} A_i; \\ RID_i^* &= h(ID_i||h(F_i^*)) \\ &\overset{?}{=} RID_i. \end{aligned}$$

If the above equations are valid, U_i produces a random number r_i and computes

$$\begin{aligned} W &= B_i \oplus h(RID_i||A_i); \\ C_1 &= r_i P; \\ C_2 &= r_i \oplus W; \\ C_4 &= h(RID_i||r_i||W||T_1). \end{aligned}$$

Here, r_i is a random number; P denotes a point on the elliptic curve; T_i denotes the current time of the smart card. Next, U_i sends messages $\{C_2, CID_i, C_4, T_1\}$ to server S.

The other Steps LA2 $\sim$ LA4 are the same as that of Liu *et al.*'s scheme.

4.3 Password Change Phase

The password change phase of the proposed scheme is shown in Figure 3. The procedures of the password change phase of the proposed scheme is listed in the following:

Step PC1: U_i starts SC. U_i keys in messages $\{ID_i, PW_i\}$, and extracts his/her fingerprint T_i. Next, U_i verifies the following parameters whether hold:

$$\begin{aligned} F_i^* &= h(T_i) \\ &\overset{?}{=} F_i; \\ A_i^* &= h(PW_i||h(F_i^*)) \\ &\overset{?}{=} A_i; \\ RID_i^* &= h(ID_i||h(F_i^*)) \\ &\overset{?}{=} RID_i. \end{aligned}$$

If the above equations are true, U_i generates a new password $PW_{i_{new}}$ and computes

$$\begin{aligned} A_{i_{new}} &= h(PW_{i_{new}}||h(F_i^*)); \\ B_{i_{new}} &= h(RID_i||A_{i_{new}}) \oplus W. \end{aligned}$$

Replaces $< A_i, B_i >$ with $< A_{i_{new}}, B_{i_{new}} >$ in Smart Card SC.

User Ui/Smartcard

PC1: Insert the smart card SC
Keys $< ID_i, PW_i >$ and Extracts T_i
Verifies
$F_i^* = H(T_i) \overset{?}{=} F_i$
$A_i^* = h(PW_i||h(F_i^*)) \overset{?}{=} A_i$
$RID_i^* = h(ID_i||h(F_i^*)) \overset{?}{=} RID_i$
If the above equations are valid,
U_i generates $PW_{i_{new}}$
Computes
$A_{i_{new}} = h(PW_{i_{new}}||h(F_i))$
$B_{i_{new}} = h(RID_i||A_{i_{new}}) \oplus W$
Replaces $< A_{i_{new}}, B_{i_{new}} >$ in SC

Figure 3: The password change phase of the proposed scheme

4.4 Cryptanalysis

In this article, we proposed an improvement of Liu *et al.* 's scheme [17]. The security analysis is similar to their scheme.

5 Conclusion

In summary, we have shown the weakness of Liu *et al.*'s user authentication scheme [17]. Their scheme could not withstand the fake smart card attack and a difficult-to-remember random number. We also propose an improved user authentication scheme that to withstand these weaknesses as that in Liu *et al.*'s scheme.

Acknowledgments

The Ministry of Science and Technology partially supported this research, Taiwan (ROC), under contract no.: MOST 109-2221-E-468-011-MY3, MOST 108-2410-H-468-023, and MOST 111-2622-8-468-001 -TM1.

References

[1] R. Amin, G. P. Biswas, "An improved RSA based user authentication and session key agreement protocol usable in TMIS," *Journal of Medical Systems*, vol. 39, no. 8, pp. 1-14, 2015.

[2] P. Annamalai, K. Raju, D. Ranganayakulu, "Soft biometrics traits for continuous authentication in online exam using ICA based facial recognition," *International Journal of Network Security*, vol. 20, no. 3, pp. 423-432, 2018.

[3] S. Q. Cao, Q. Sun, L. L. Cao, "Security analysis and enhancements of a remote user authentication scheme," *International Journal of Network Security*, vol. 21, no. 4, pp. 661-669, 2019.

[4] C. C. Chang, C. Y. Lee, "A smart card-based authentication scheme using user identity cryptography," *International Journal of Network Security*, vol. 16, pp. 139-147, 2013.

[5] T. Y. Chang, W. P. Yang, M. S. Hwang. , "Simple authenticated key agreement and protected password change protocol," *Computers & Mathematics with Applications*, vol. 49, pp. 703-714, 2005.

[6] T. Y. Chen, C. C. Lee, M. S. Hwang, J. K. Jan, "Towards secure and efficient user authentication scheme using smart card for multi-server environments," *The Journal of Supercomputing*, vol. 66, no. 2, pp. 1008-1032, 2013.

[7] S. F. Chiou, E. F. Cahyadi, C. Y. Yang, M. S. Hwang, "An improved Chang-Lee's smart card-based authentication scheme," *Journal of Physics: Conference Series*, ICSP 2019, 2019.

[8] S. Y. Chiou, W. T. Ko, E. H. Lu, "A secure ECC-based mobile RFID mutual authentication protocol and its application," *International Journal of Network Security*, vol. 20, no. 2, pp. 396-402, 2018.

[9] S. F. Chiou, H. T. Pan, E. F. Cahyadi, M. S. Hwang. , "Cryptanalysis of the mutual authentication and key agreement protocol with smart cards for wireless communications," *International Journal of Network Security*, vol. 21, no. 1, pp. 100-104, 2019.

[10] R. H. Dong, B. B. Ren, Q. Y. Zhang, H. Yuan, "A lightweight user authentication scheme based on fuzzy extraction technology for wireless sensor networks," *International Journal of Network Security*, vol. 23, no. 1, pp. 157-171, 2021.

[11] C. Guo, C. C. Chang, S. C. Chang, "A secure and efficient mutual authentication and key agreement protocol with smart cards for wireless communications," *International Journal of Network Security*, vol. 20, no. 2, pp. 323-331, 2018.

[12] L. Han, Q. Xie, W. Liu, "An improved biometric based authentication scheme with user anonymity using elliptic curve cryptosystem," *International Journal of Network Security*, vol. 19, no. 3, pp. 469-478, 2017.

[13] G. Hou, Z. Wang, "A robust and efficient remote authentication scheme from elliptic curve cryptosystem," *International Journal of Network Security*, vol. 19, no. 6, pp. 904-911, 2017.

[14] M. S. Hwang, L. H. Li, "A new remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28-30, 2000.

[15] J. Liu, X. He, H. Tang, D. Wang, B. Meng, "A novel privacy-preserving user authentication protocol for big data environment," *International Journal of Network Security*, vol. 23, no. 3, pp. 436-448, 2021.

[16] L. Liu, L. Hong, Z. Cao, "Analysis of one secure key agreement and key protection for mobile device user authentication," *International Journal of Network Security*, vol. 24, no. 2, pp. 238-242, 2022.

[17] W. R. Liu, X. He, Z. Y. Ji, "Security analysis and enhancements of a user authentication scheme," *International Journal of Network Security*, vol. 23, pp. 895-903, 2021.

[18] W. R. Liu, B. Li, Z. Y. Ji, "An improved three-factor remote user authentication protocol using elliptic curve cryptography," *International Journal of Network Security*, vol. 24, no. 3, pp. 521-532, 2022.

[19] Y. Liu, C. C. Chang, P. C. Huang, "One-code-pass user authentication based on QR code and secret sharing," *International Journal of Network Security*, vol. 22, no. 5, pp. 752-762, 2020.

[20] J. Mo, Z. Hu, "Comments on a remote user authentication scheme for multi-server 5G networks," *International Journal of Network Security*, vol. 23, no. 5, pp. 878-882, 2021.

[21] J. Moon, D. Lee, J. Jung, D. Won, "Improvement of efficient and secure smart card based password authentication scheme," *International Journal of Network Security*, vol. 19, pp. 1053-1061, 2017.

[22] J. Saadatmandan, A. Rahimi, "Digital certificate of public key for user authentication and session key establishment for secure network communications," *International Journal of Network Security*, vol. 23, no. 3, pp. 480-489, 2021.

[23] J. J. Shen, C. W. Lin, M. S. Hwang. , "A modified remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 2, pp. 414-416, 2003.

[24] E. Tarek, O. Ouda, A. Atwan, "Image-based multimodal biometric authentication using double random phase encoding," *International Journal of Network Security*, vol. 20, no. 6, pp. 1163-1174, 2018.

[25] C. S. Tsai, C. C. Lee, M. S. Hwang, "Password authentication schemes: Current status and key issues.," *International Journal of Network Security*, vol. 3, pp. 101-115, 2006.

[26] C. H. Wei, M. S. Hwang, A. Y. H. Chin, "A mutual authentication protocol for RFID," *IEEE IT Professional*, vol. 13, no. 2, pp. 20-24, 2011.

[27] C. C. Yang, T. Y. Chang, M. S. Hwang. , "The security of the improvement on the methods for protecting password transmission," *Informatica*, vol. 14, pp. 551-558, 2003.

[28] H. W. Yang, H. T. Pan, Y. H. Chen, M. S. Hwang, "A Taxonomy of user authentication schemes for multi-server environments," *International Journal of Network Security*, vol. 22, no. 3, pp. 365-372, 2020.

[29] Q. Zhang, J. Zhang, L. Liu, J. Wang, P. Liu, "On security of privacy-preserving remote user authentication with k-times untraceability," *International Journal of Network Security*, vol. 23, no. 3, pp. 449-454, 2021.

Biography

Min-Shiang Hwang received M.S. in industrial engineering from National Tsing Hua University, Taiwan in

1988, and a Ph.D. degree in computer and information science from National Chiao Tung University, Taiwan, in 1995. He was a distinguished professor and Chairman of the Department of Management Information Systems, National Chung Hsing University, during 2003-2011. He obtained 1997, 1998, 1999, 2000, and 2001 Excellent Research Awards from the National Science Council (Taiwan). Dr. Hwang was a dean of the College of Computer Science, Asia University (AU), Taichung, Taiwan. He is currently a chair professor with the Department of Computer Science and Information Engineering, AU. His current research interests include information security, electronic commerce, database, data security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 300+ articles on the above research fields in international journals.

Hou-Wen Li received a bachelor's degree in business administration from National Cheng Kung University in 1992 and a master's degree in law from Tunghai University in Taiwan in 2009.He worked as a teacher at the New Taipei Municipal New Taipei Industrial Vocational High School from 1992 to 2002, and at the National Taiwan University of Sport since 2002. He had worked on the Regulations Committee and is a professional investigator of gender equality for the Ministry of Education. He is currently pursuing his Ph.D. in the Artificial Intelligence Ph.D. program at Asia University. His research interests include natural language processing and the application of artificial intelligence in law, etc.

Cheng-Ying Yang received the M.S. degree in Electronic Engineering from Monmouth University, New Jersey, in 1991, and Ph.D. degree from the University of Toledo, Ohio, in 1999. He is a member of IEEE Satellite & Space Communication Society. Currently, he is employed as a Professor at Department of Computer Science, University of Taipei, Taiwan. His research interests are performance analysis of digital communication systems, error control coding, signal processing and computer security.

Image Encryption Based on Hyperchaotic Systems And DNA Encoding

Lei Wang
(Corresponding author: Lei Wang)

Basic Department, Zhengzhou University of Science and Technology
Email: chenwwencww@163.com

(Received Dec. 25, 2022; Revised and Accepted Apr. 16, 2023; First Online Apr. 30, 2023)

The Special Issue on Computational Intelligence Networks for Privacy and Security in Evolving Internet of Multimedia Things
Special Editor: Prof. Shoulin Yin (Harbin Institute of Technology)

Abstract

To effectively improve the quality of image encryption and the security of data transmission, this paper proposes a new image encryption theme based on Hyperchaotic systems and DNA encoding. First, L plaintext images are operated by a double-layer cross-coupled piece-wise linear chaotic map (PWLCM). And the noise-like images are obtained by XOR merging. Then, based on the chaotic sequence, the image plaintext is encoded by DNA, and the image scrambling and diffusion are realized by DNA operation, so the image encryption is completed. Finally, Simulation results show that this new encryption algorithm's number of Pixels Change Rate (NPCR) and Unified Average Change Intensity (UACI) is close to or higher than the theoretically calculated values. Furthermore, the Peak Signal Noise Ratio (PSNR) is less than 10, indicating that the proposed algorithm is susceptible to plaintext and key and can effectively resist differential attacks.

Keywords: DNA Encoding; Hyperchaotic Systems; Image Encryption; Scrambling and Diffusion

1 Introduction

With the development of network, there are many information carriers. Digital image has become the most extensive communication medium in network communication because of its high fidelity and vivid image. But in the actual network life and real life, there are a lot of images are used fraudulantly, copyright infringed [5, 15, 16]. For example, some people will use other people's photos to carry out illegal activities, violating others' privacy. No one wants their privacy violated. Therefore, the security of image information becomes the focus of people's attention, and some image encryption technology is urgently needed to protect the image information.

Image information redundancy is large. Generally for text encryption practical encryption algorithms such as DES [21], 3DES [2] and so on, it cannot achieve good results on the image, therefore, chaotic encryption is arised. Since many earlier chaotic systems are easy to be attacked and cracked, resulting in information exposure. So high-dimensional chaotic systems, hyperchaotic systems, deep learning, wavelet transform combined with chaotic system image encryption operations have been emerged.

Chaos is widely used in the field of image encryption because of its randomness, high sensitivity to initial value and replicability [20]. There are two kinds of image encryption methods. One is to improve the original chaotic mapping to increase its complexity and security. The second is to improve the encryption algorithm. Reference [17] expanded the scope of the mapping by improving the Logistic mapping. Reference [6] increased the complexity of sequences and enhanced the scrambling diffusion effect by improving Henon mapping. Reference [18] improved Lorenz equation to make its chaotic behavior more complicated. Reference [24] combined Sine mapping and Logistic mapping to form a chaotic system with higher complexity.

Traditional encryption algorithms, such as Arnold scrambling transform [8] and Hilbert scrambling transform [14], have some defects and low security. Therefore, reference [12] proposed an encryption algorithm that scrambled filling curves and adjacent pixel bits. Reference [23] proposed a hybrid Encryption algorithm based on the advantages of Data Encryption Standard (DES) algorithm, such as high efficiency, strong security and good performance, combined with Logistic mapping. Reference [4] proposed the calculation of Deoxyribo Nucleic Acid (DNA), which provided a new direction for image encryption algorithm technology. In reference [10], two-dimensional Logistic mapping was used to generate chaotic sequences, and DNA coding algorithm was combined to encrypt images. In reference [19], Lorenz and Rossler double chaotic system was used by combining with DNA coding to increase the complexity of computation and difficulty of cracking. In this paper, an improved Hyperchaotic chaotic map with better chaos is designed.

An image encryption algorithm based on Hyperchaotic systems and DNA encoding is proposed by combining with the DNA sequences with high parallelism and abundant storage space.

2 Related Works

2.1 Piece-wise Linear Chaotic Map (PWLCM)

When selecting any chaotic mapping in image encryption, two important characteristics of chaotic mapping, namely "simplicity" and "ergodicity", must be considered. Compared with other one-dimensional chaotic systems, PWLCM is relatively uniform in phase distribution and has simple equations, satisfying the above two features [3]. Therefore, the PWLCM system will be used in this paper to generate random sequences, its dynamic equation is defined as follows:

$$x_{i+1} = F_p(x_i) \quad = \quad \left\{ \begin{array}{l} x_i/p \text{ if } 0 \leq x_i < p \\ x_i - p/0.5 - p \text{ if } p \leq x_i < 0.5 \\ F_p(1 - x_i) \ 0.5 \leq \mathrm{x_i} < 1 \end{array} \right\}$$

Where, p is the control parameter, whose value range is $(0, 0.5)$. $x_i \in [0, 1)$ is the state variable. In the encryption algorithm in this paper, in order to obtain a more unpredictable chaotic sequence, PWLCM mapping is used to carry out double-layer cross-coupling operation, and the behavior trajectory generated is more complex and not easy to predict, which can achieve a better image scrambling effect.

2.2 2D-CTMM Chaotic System

Low-dimensional chaotic system runs fast, but it has some problems, such as small key space, easy to predict behavior trajectory and low security performance. However, the behavior trajectory of high-dimensional chaotic system is difficult to predict and the structure is complex, which leads to the decrease of encryption rate. After weighing encryption rate and encryption security, this paper combines one-dimensional tent chaotic mapping [9] and one-dimensional cubic chaotic mapping to propose two-dimensional cubic tent chaotic mapping (2D-CTMM). This is a new chaotic system, which combines two one-dimensional chaotic systems. Compared with other high-dimensional chaotic systems, 2D-CTMM has a simple structure. Compared with low-dimensional chaotic system, its behavior trajectory is not easy to predict. On the premise that encryption security is satisfied, 2D-CTMM has a relatively high running rate, and its system equation is shown in Equations (1) and (2):

$$x_{i+1} \quad = \quad \left\{ \begin{array}{l} 4ax_i + 4by_i/0.5mod1 \text{ if } y_i < 0.5 \\ 4ax_i + 4b(1 - y_i)/0.5mod1 \text{ if } y_i \geq 0.5 \end{array} \right. \quad (1)$$

$$y_{i+1} \quad = \quad \left\{ \begin{array}{l} 4ay_i + 4bx_i/0.5mod1 \text{ if } x_i < 0.5 \\ 4ay_i + 4b(1 - x_i)/0.5mod1 \text{ if } x_i \geq 0.5 \end{array} \right. \quad (2)$$

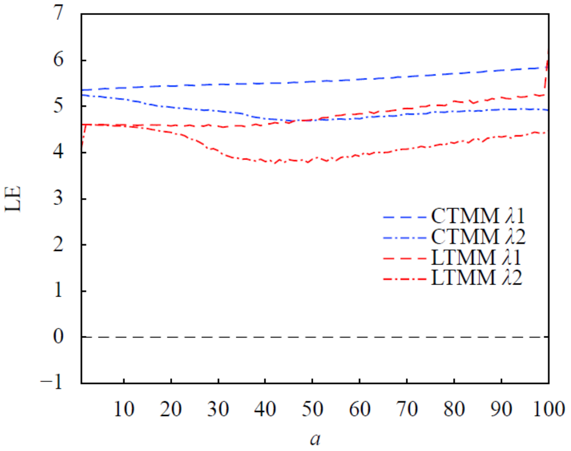


Figure 1: Comparison of LE curve between CTMM and LTMM

Where, a and b are control parameters of the 2D-CTMM system. *mod* is complementary function. Since the modular operation of a 2D-CTMM chaotic system is whole-office bounded, it can always fold the value into a fixed range, so the value of the control parameter can be set to any large value. In this article, the parameter range is set to $a, b \in [1, 100]$.

2.3 Performance Analysis of 2D-CTMM Chaotic System

Lyapunov exponent (LE) is a key quantitative index to measure the dynamic characteristics of the system, which describes the convergence rate or divergence rate of the system trajectory. When there are multiple Lyapunov exponentials greater than zero in a chaotic system, it indicates that the chaotic system has hyperchaotic behavior.

Compared with other two-dimensional chaotic maps, Two-dimensional logistic tent modular map (2D-LTMM) shows better chaotic characteristics, so in this paper, the Lyapunov exponential curve of 2D-CTMM and 2D-LTMM is compared, as shown in Figure 1. The initial value is set as $x0 = 0.528$, $y0 = 0.135$, control parameter $b = 50$, $a \in [1, 100]$. As can be seen from Figure 1, 2D-CTMM is in hyperchaotic behavior in the whole interval range, and compared with 2D-LTMM, 2D-CTMM has a larger LE value, indicating that it has more complex chaotic characteristics.

2.4 DNA Encoding Rule and Operation

According to the base-complementary pairing rules in biology, adenine (A) complements thymine (T) and cytosine (C) complements guanine (G). This is similar to the complementarity of 0 and 1 in binary, with the binary number being 00, 01, 10 and 11. There are 24 encoding rules according to permutation and combination, but only 8 encoding rules in Table 1 are left according to DNA encoding rule [22]. The operations of DNA sequence mainly include addition, subtraction and XOR operations. Eight kinds of rules correspond to eight kinds of DNA arithmetic rules.

Table 1: DNA encoding rule

DNA type	A	T	C	G
1	00	11	01	10
2	00	11	10	01
3	01	10	00	11
4	01	10	11	00
5	10	01	00	11
6	10	01	00	11
7	11	01	11	10
8	11	00	10	01

3 Proposed Image Encryption Algorithm

Known plaintext matrix P and the chaotic sequence generated by the key set $A = A_1, A_2, \cdots, A_I | I \in C^*$, $B = B_1, B_2, \cdots, B_I | I \in C^*$. Where A_I and B_I represent the $I - th$ chaotic sequence. C^* represents the set of positive integers. In this paper, the encryption process of the image encryption algorithm based on Hyperchaotic systems and DNA encoding is shown in Figure 2. First, it takes each element in sets A and B to 8 decimal places, and maps the element in set A to [1, 8] through certain operations to obtain $Q = Q_1, Q_2, \cdots, Q_I | I \in C^*$. Similarly, it maps the elements of set B to [1,8] and [1,3] respectively, and gets $W = W_1, W_2, \cdots, W_I | I \in C^*$ and $E = E_1, E_2, \cdots, E_I | I \in C^*$. Q and W correspond to 8 encoding rules, and E corresponds to 3 operation rules. Secondly, the elements in set A are converted into a ciphertext matrix of the same size $(M \times N)$ as the plaintext matrix P. Each element in the plain-text and ciphertext matrices is converted to an 8-bit binary number. According to the DNA encoding rules in Table 1, DNA encoding is performed for every 2 bits of binary number. If the plaintext encoding rule is determined by Q and the ciphertext encoding rule is determined by W, each element can be converted into four DNA encodes. In order to ensure that the matrix size remains $M \times N$ after DNA encoding, the plaintext and ciphertext are partitioned according to every 4 DNA codes to generate a new plaintext matrix P' and a new ciphertext matrix A' with size $M \times N$ and composed of DNA codes. Then, DNA operation is carried out on DNA code blocks at corresponding positions in the new plaintext matrix and the new ciphertext matrix. The operation rules are determined by E, and the scrambling matrix R is obtained. Next, starting with the last element, each element in the scrambled matrix R is DNA computed with the previous element to obtain the diffusion matrix R'. Finally, the diffusion matrix is decoded for DNA and restored to binary sequence, and the final encrypted image matrix R'' is obtained by reconstructing the matrix.

3.1 Key Generation

In the process of generating the system key, this paper uses two improved mappings. The first mapping produces the initial values x_0, y_0 and parameters a, b of the chaotic sequence. The second mapping produces the initial values x'_0, y'_0 and parameters a', b' of the chaotic sequence. The generating process of initial values x_0, x'_0, y_0, y'_0 is connected with the original image information to form a dynamic key and achieve an one-password encryption effect. Parameters a, a', b, b' are used as fixed keys.

Firstly, XOR operation is performed by pixel and an 8-bit binary number, then all pixel values are added to generate a new value, and then divided by the size of the plaintext image. Finally, the decimal part is taken as the key. The calculation formula is as follows:

$$K_i = mod(sum(P \oplus N_i)/(M \times N), 1).$$

Where, K is the generated key and P is the plaintext information. N_i is any number in the range of 0 to 255. mod indicates mod operation. $M \times N$ is the size of the plaintext image.

3.2 Generation of Decision Parameters

The image encryption algorithm based on hyperchaotic sequence and DNA encoding in this paper has three decision parameters, Q, W and E. Q is used to determine the DNA encoding mode of the plaintext and the final decoding mode, which is generated by the first chaotic sequence. W is used to determine the DNA encoding mode of the ciphertext. E is used to determine the algorithm and is generated by the second chaotic sequence.

Firstly, it takes the decimal part of the sequence, and then converts the sequence value to 0-255. The calculation formula is as follows:

$$\begin{aligned} A'(i) &= A(i) - floor(A(i)). \\ A''(i) &= mod(floor(A'(i) \times 10^8), 256). \end{aligned}$$

Then the parameters Q, W and E are generated. The size of Q and W ranges from 1 to 8, corresponding to 8 encoding rules. The range of E is 1-3 corresponding to 3 operation modes, and the values of $Q(i)$, $W(i)$ and $E(i)$ are converted into:

$$\begin{aligned} Q(i) &= \text{mod}(A''(i), 8) + 1. \\ W(i) &= \text{mod}(B''(i), 8) + 1. \\ E(i) &= \text{mod}(B''(i), 3) + 1. \end{aligned}$$

3.3 Scrambling and Diffusion Operations

First, the elements in the plaintext and ciphertext matrices are converted to binary numbers. According to the

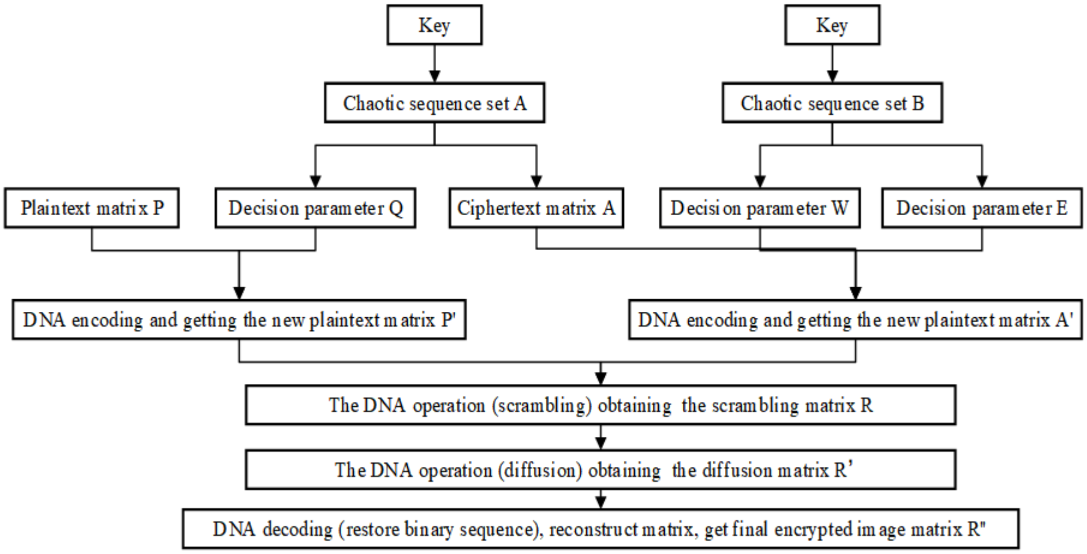

Key
Chaotic sequence set A
Key
Chaotic sequence set B
Plaintext matrix P
Decision parameter Q
Ciphertext matrix A
Decision parameter W
Decision parameter E
DNA encoding and getting the new plaintext matrix P'
DNA encoding and getting the new plaintext matrix A'
The DNA operation (scrambling) obtaining the scrambling matrix R
The DNA operation (diffusion) obtaining the diffusion matrix R'
DNA decoding (restore binary sequence), reconstruct matrix, get final encrypted image matrix R"

Figure 2: Image encryption process

encoding rules in Table 1, every 2 bits are used for DNA encoding, and every 4 DNA codes form an encoding block. The plaintext selects the encoding rule according to the Q value, and the ciphertext selects the encoding rule according to the W value, and each element corresponds to different Q and W, realizing dynamic encoding. Then the scrambling operation is performed. According to the corresponding E value, DNA operation is performed on the plaintext block and ciphertext block. The calculation formula is as follows:

$$R(i) = f(A(i), P(i), E(i)).$$

Where, $R(i)$ is the block of the $i-th$ block after scrambling. f is DNA operation. $A(i)$ is the $i-th$ ciphertext block after DNA encoding. $P(i)$ is the $i-th$ plaintext block after DNA encoding. $E(i)$ is the operation mode selected by block i.

Then there is the diffusion operation. DNA operation is performed again on $R(i)$ and the previous scrambled block $R(i-1)$. The calculation method is also determined by the value of E. The calculation formula is as follows:

$$R'(i) = f(R(i), R(i-1), E(i)).$$

4 Image Encryption Performance Analysis

4.1 Encryption Effect and Histogram Analysis

The dynamic key $x_0 = 0.8945$, $y_0 = 0.3694$, $x'_0 = 0.9978$, $y'_0 = 0.3642$ is generated by the improved hyperchaotic mapping. Fixed key $a = 12.0011$, $b = 40.0012$, $a' = 16.3779$, $b' = 42.8676$. A 256-level gray image of 512×512 is selected and MATLAB2020b platform is adopted for simulation experiment. The image encryption effect and histogram are shown in Figures 3 and 4.

Image decryption is the reverse process of encryption. First, two chaotic sequences are generated using the eight keys used in encryption, and the decision parameters, DNA decoding, and inversion rules are generated from them. That is, DNA addition is decrypted by subtraction, and subtraction is decrypted by addition. Then, according to DNA inversion rules and decision parameters, the diffusion and scrambling operations are carried out successively. Finally, DNA decoding is used to restore the binary sequence to get the decrypted image. The image decryption effect and histogram are shown in Figures 3 & 4. As can be seen from figure 3, the decrypted image is completely consistent with the original image and its histogram after decryption using the proposed algorithm, indicating that the proposed algorithm has a good decryption effect. As can be seen from Figure 4, the encrypted image can no longer distinguish the original image information visually and is close to the noisy image. The gray value distribution of encrypted images is more uniform, which means that the images are more difficult to identify, provide less effective information, and have higher security.

4.2 Information Entropy Analysis

Information entropy [11] is one of the indicators to measure the effect of image encryption. The maximum entropy of a grayscale image is 8. Ifthe encryption effect is better, the information entropy is closer to 8. The calculation formula is as follows:

$$H(x) = -\sum_{i=1}^{2N-1} P_i log_2 P_i.$$

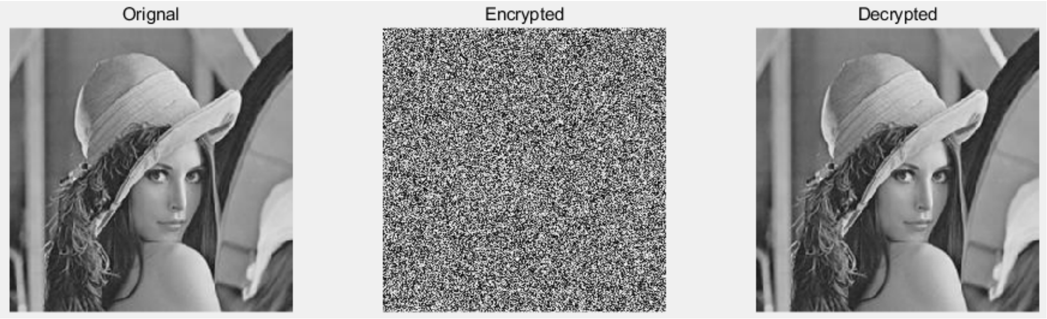


Figure 3: Encryption and decryption effect

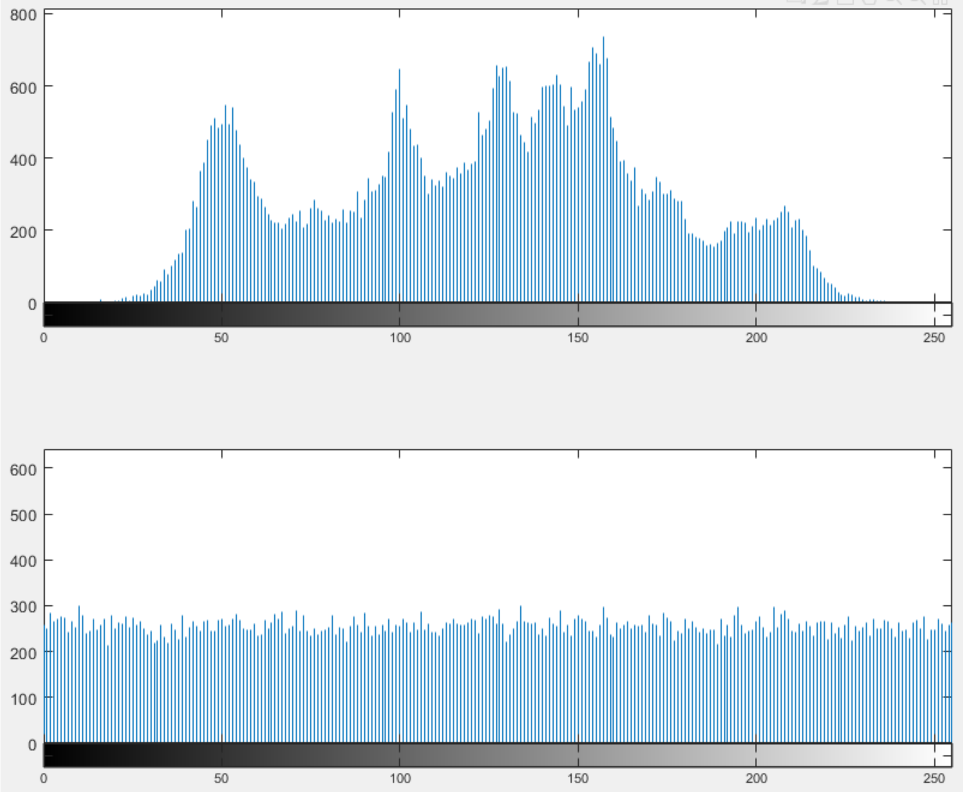


Figure 4: Encryption and decryption image gray histogram

Where, $H(x)$ is information entropy. P_i is the probability of gray value i.

The plaintext image in figure 3 is selected and the ciphertext image is encrypted by the algorithm in this paper. The entropy of plaintext and ciphertext information is calculated and compared with the entropy values obtained in reference [7] and reference [13]. The results are shown in Table 2.

Table 2: Plaintext and ciphertext entropy of different algorithms

Method	Plaintext entropy	Ciphertext entropy
Proposed	7.3733	7.9995
reference [7]	7.4543	7.9974
reference [13]	7.3451	7.9896

As can be seen from Table 2, in the three algorithms, the image information entropy after encryption with the proposed algorithm is closer to 8, indicating that the proposed algorithm has better encryption effect.

4.3 Differential Attack Resistance and Sensitivity Analysis

Differential attack is mainly through changing the original image information, and then encrypting with the same key by comparing the difference before and after the image encryption to find the difference between the plaintext and the key, so as to decipher the encrypted image. Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are used to evaluate the differential attack resistance capability. NPCR reflects the ratio of different gray values of different images at the same position. UACI reflects the average density of change between different images. The larger NPCR and UACI denote the better differential attack resistance, the stronger sensitivity, and the better encryption effect. NPCR and UACI are calculated as follows:

$$NPCR = \frac{\sum_{i,j} G(i,j)}{M \times N} \times 100/\%.$$

$$UACI = \frac{1}{M \times N} \sum_{i,j} \frac{R_1(i,j) - R_2(i,j)}{255} \times 100/\%.$$

In the formula, $R_1(i,j)$ and $R_2(i,j)$ are the pixel values of pixel points in $i-th$ row and $j-th$ column of the original encrypted image and the changed encrypted image respectively. When $R_1(i,j) = R_2(i,j)$, $G(i,j) = 0$, otherwise $G(i,j) = 1$.

In reference [1], the 256-grade gray image was theoretically calculated according to Equations (**??**) and (**??**), and the theoretical calculated values of NPCR and UACI were 99.6094% and 33.465%, respectively.

In the experiment of plaintext sensitivity analysis, firstly, the new algorithm in this paper is used to encrypt the original plaintext image to form the original encrypted image. Then, the pixel value of a certain point in the original plaintext image is changed, and the new algorithm is used for encryption to form a new encrypted image. The original encrypted image is compared with the new encrypted image to obtain NPCR and UACI. In the experiment, $P(i,j)$ represents the pixel value of coordinate point (i,j). The pixel value of the coordinate (20,30) is changed from 12 to 13, the pixel value of the coordinate (155,100) is changed from 14 to 15, the pixel value of the coordinate (200,300) is changed from 169 to 170, and the pixel value of the coordinate (512,512) is changed from 65 to 66. The test results are shown in Table 3.

Table 3: Plaintext sensitivity of the proposed algorithm

Index	NPCR	UACI
P(20,30)	99.6151	33.4862
P(155,100)	99.6372	33.4015
P(200,300)	99.6179	33.4062
P(512,512)	99.6234	33.4323

As can be seen from Table 3, the calculated NPCR and UACI values are close to or higher than the theoretical calculated values in reference [1] when the pixel value of a coordinate point of the plaintext image is slightly changed and encrypted by the new algorithm in this paper. It shows that the proposed algorithm can effectively resist differential attacks and has high plaintext sensitivity.

5 Conclusion

This paper proposes an image encryption algorithm based on hyperchaotic mapping and DNA coding, which enhances the security of images, expands the key space, and improves the ability to resist differential attacks. However, the new algorithm in this paper only applies to 256-level gray image encryption. The subsequent plan is to carry out research on color image encryption, extract RGB channels of color images, and carry out chaotic encryption for each channel and between channels, so as to further improve the application of the algorithm in the field of image encryption.

Acknowledgments

This study was supported by the Key Project of Science and Technology Department of Henan Province, Project No. 172102210113. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] H. N. Abdullah, H. A. Abdullah, "Image encryption using hybrid chaotic map," *2017 International*

Conference on Current Research in Computer Science and Information Technology (ICCIT). IEEE, pp. 121-125, 2017.

[2] R. P. Adhie, Y. Hutama, A. S. Ahmar, *et al.*, "Implementation cryptography data encryption standard (DES) and triple data encryption standard (3DES) method in communication system based near field communication (NFC)," *Journal of Physics: Conference Series. IOP Publishing*, vol. 954, no. 1, pp. 012009, 2018.

[3] A. Ali, M. A. Khan, R. K. Ayyasamy, *et al.*, "A novel systematic byte substitution method to design strong bijective substitution box (S-box) using piece-wise-linear chaotic map," *PeerJ Computer Science*, vol. 8, pp. e940, 2022.

[4] H. R. Amani, M. Yaghoobi, "A new approach in adaptive encryption algorithm for color images based on DNA sequence operation and hyper-chaotic system," *Multimedia Tools and Applications*, vol. 78, pp. 21537-21556, 2019.

[5] C. C. Chang, K. F. Hwang, M. S. Hwang, "Robust authentication scheme for protecting copyrights of images and graphics", *IEE Proceedings-Vision, Image and Signal Processing*, vol. 149, no. 1, pp. 43-50, 2002.

[6] Y. Chen, S. Xie, J. Zhang, "A hybrid domain image encryption algorithm based on improved henon map," *Entropy*, vol. 24, no. 2, pp. 287, 2022.

[7] M. Gupta, V. P. Singh, K. K. Gupta, *et al.*, "An efficient image encryption technique based on two-level security for internet of things," *Multimedia Tools and Applications*, vol. 82, no. 4, pp. 5091-5111, 2023.

[8] H. Huang, Z. Cai, "Duple color image encryption system based on 3D non-equilateral arnold transform for IIoT," *IEEE Transactions on Industrial Informatics*, 2022.

[9] S. Kanwal, S. Inam, M. T. B. Othman, *et al.*, "An effective color image encryption based on henon map, tent chaotic map, and orthogonal matrices," *Sensors*, vol. 22, no. 12, pp. 4359, 2022.

[10] H. Liu, B. Zhao, L. Huang, "A remote-sensing image encryption scheme using DNA bases probability and two-dimensional logistic map," *IEEE Access*, vol. 7, pp. 65450-65459, 2019.

[11] Y. Liu, T. Zhi, M. Shen, *et al.*, "Software-defined DDoS detection with information entropy analysis and optimized deep learning," *Future Generation Computer Systems*, vol. 99-114, no. 129, 2022.

[12] P. Praveenkumar, R. Amirtharajan, K. Thenmozhi, *et al.*, "Fusion of confusion and diffusion: A novel image encryption approach," *Telecommunication Systems*, vol. 65, no. 65-78, 2017.

[13] Y. Sang, J. Sang, M. S. Alam, "Image encryption based on logistic chaotic systems and deep autoencoder," *Pattern Recognition Letters*, vol. 153, pp. 59-66, 2022.

[14] V. K. Sharma, P. C. Sharma, H. Goud, *et al.*, "Hilbert quantum image scrambling and graph signal processing-based image steganography," *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 17817-17830, 2022.

[15] A. Shobanadevi, G. Maragathm, S. M. P. Gangadharan, *et al.*, "Internet of Things-based data hiding scheme for wireless communication," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1-8, 2022.

[16] W. Wan, J. Wang, Y. Zhang, *et al.*, "A comprehensive survey on robust image watermarking," *Neurocomputing*, Vol. 488,?pp. 226-247, 2022.

[17] H. Xiang, L. Liu, "An improved digital logistic map and its application in image encryption," *Multimedia Tools and Applications*, vol. 79, pp. 30329-30355, 2020.

[18] L. Wang, H. Song, P. Liu, "A novel hybrid color image encryption algorithm using two complex chaotic systems," *Optics and Lasers in Engineering*, vol. 77, pp. 118-125, 2016.

[19] S. Wang, L. Hong, J. Jiang, "An image encryption scheme using a chaotic neural network and a network with multistable hyperchaos," *Optik*, vol. 268, pp. 169758, 2022.

[20] X. Wang, S. Yin, M. Shafiq, A. A. Laghari, S. Karim, O. Cheikhrouhou, W. Alhakami, H. Hamam, "A new V-net convolutional neural network based on four-dimensional hyperchaotic system for medical image encryption," *Security and Communication Networks*, vol. 2022, pp. 1-14, 2022.

[21] W. Yihan, L. Yongzhen, "Improved design of DES algorithm based on symmetric encryption algorithm," *2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA). IEEE*, pp. 220-223, 2021.

[22] S. Yin, H. Li, "GSAPSO-MQC:medical image encryption based on genetic simulated annealing particle swarm optimization and modified quantum chaos system," *Evolutionary Intelligence*, vol. 14, pp. 1817-1829, 2021.

[23] Q. Zhang, "An overview and analysis of hybrid encryption: The combination of symmetric encryption and asymmetric encryption," in *2nd international conference on computing and data science (CDS)*, IEEE, pp. 616-622, 2021.

[24] J. Zheng, L. F. Liu, "Novel image encryption by combining dynamic DNA sequence encryption and the improved 2D logistic sine map," *IET image processing*, vol. 14, no. 11, pp. 2310-2320, 2020.

Biography

Lei Wang biography. Wang Lei (1980-), Associate professor, Zhengzhou University of Science and Technology, born in Nanyang, Henan Province, 450064, China. He is mainly engaged in differential equation theory and algorithm research.

HWKA: A Novel Homomorphic Encryption and k-center Algorithm for Secure Storage of English Data

Haiying Liu

(Corresponding author: Haiying Liu)

School of Foreign Languages, Zhengzhou University of Science and Technology

Email: ldadahai@163.com

(Received Dec. 28, 2022; Revised and Accepted Apr. 16, 2023; First Online Apr. 30, 2023)

The Special Issue on Computational Intelligence Networks for Privacy and Security in Evolving Internet of Multimedia Things

Special Editor: Prof. Shoulin Yin (Harbin Institute of Technology)

Abstract

Aiming at the security problem of English private data in a cloud environment, this paper puts forward a novel homomorphic encryption and k-center algorithm for the secure storage of English data. Firstly, we cluster the English data. The clustering method is improved by using the clustering process under the condition of changing the vertical data distribution. Secondly, Paillier homomorphic encryption is introduced in clustering. Then, this method is applied to the k-center clustering algorithm, which further guarantees data security. The experimental results show that the efficiency of the ciphertext computing model is greatly improved. Moreover, the model puts a lot of computing in the cloud environment, which can reduce the client's pressure and fully use the resources in the cloud environment.

Keywords: Clustering; Data Secure Storage; Homomorphic Encryption; k-center Algorithm

1 Introduction

At present, more and more enterprises and individuals begin to use cloud storage to store data and use cloud computing to process data [7,9,10,16]. Cloud computing is easy to expand, it has low requirements on devices, and enhanced computing power, which improves resource utilization and reduces costs [5,11]. Moreover, the large-scale cluster and huge computing power of cloud environment enable cloud computing to process big data, carry out data mining on big data, and realize classification, clustering and image recognition through machine learning algorithm [2].

Generally, the structure of machine learning for privacy protection can be divided into two categories: privacy training and privacy classification. Existing studies focus on the first type, which not only protects the privacy of the training samples provided by the data provider, but also protects the classifier parameters of the evaluator and the prediction results of the client [20], that is, only the client can obtain the prediction results.

There are many algorithms for privacy data protection, such as naive Bayes, decision trees, linear discriminant classifiers and more general kernel methods. Reference [21] proposed a back-propagation neural network training algorithm that was suitable for randomly segmenting training data sets and protecting privacy. Reference [18] proposed to use a single homomorphic encryption scheme to train multiple machine learning classifiers. Reference [8] used parallel deep learning to design, implement and evaluate a deep learning for privacy protection. In the above studies, each participant trained the local data set with the same neural network model and used the sharing of the model's selective parameters as a technique to benefit from other participants' models without explicitly sharing training inputs. But this approach took up storage space for the participants who keep sensitive data [17]. At the same time, each participant had to go through some calculations to train.

With the deepening of research, homomorphic encryption algorithms begin to improve the resistance against attack algorithms. Indiscriminability (IND) and nonmalleability [15] are the most important things for encryption schemes because they are contradictory. In this paper, the definition of security objective of involved encryption scheme refers to indiscriminability. For all homomorphic encryption schemes, it has homomorphic properties that make it impossible for any of the all homomorphic encryption schemes to fulfill the security requirements of IND-CCA2. In the existing researches, the security of all homomorphic encryption is mostly considered from IND-CPA. In 2010, Loftus et al. implemented the first fully homomorphic encryption scheme with IND-CCA1 security [12]. Akleylek et al. [1] used "Modified Key" and

"Modified Decryption" to attack the homomorphic encryption scheme, and completed the attack on Gentry's encryption scheme. Hu's scheme could decrypt ciphertexts in specific subsets in Gentry's ciphertext space.

For the homomorphic encryption scheme, because it can realize homomorphic calculation of ciphertext, the attacker can access the decryption predictor after the ciphertext is given. Chen et al. [4] proposed a feedback attack algorithm to solve this problem. In their research, they assumed that the attacker was a cloud server, which expected to obtain users' private data while performing operations. If the plaintext space was set as $(0, 1)$, the attacker would add their selected ciphertext when returning the user ciphertext. After the user decrypted and calculated, the attacker would complete the feedback attack based on the observed result [14].

How to protect the privacy and security of English data while maintaining better data mining has become an important challenge. In this paper, the clustering method is improved by changing the clustering process under the condition of vertical distribution of data, and homomorphic encryption technology is introduced in the clustering. Then this method is used in the k-center clustering algorithm, so that the security of data is further guaranteed.

2 Preliminaries

2.1 k-center Algorithm

The description of cluster analysis is as follows. Set a data set that needs cluster analysis as $S = (S_1, S_2, \cdots, S_n)$. In this data set, each sample S is composed of its characteristic data into a vector with m dimensions $(S_{i1}, S_{i2}, \cdots, S_{im})$. When clustering S, A_i is one of the clusters, and all clusters satisfy the condition $U_{i=1}^{t} A_t = S$, and $A_i \cap A_j = \phi(i \neq j)$. Cluster analysis can be divided into static clustering and dynamic clustering. Static clustering refers to the fact that the number of clusters has been determined before the start of clustering, time t is a definite value. In dynamic clustering, the number of clusters is not determined in advance, but it is based on the actual situation of the sample data set. In the era of data information explosion, cluster analysis deals with a huge amount of data, and has a variety of forms. The traditional cluster analysis technology has been greatly challenged. These problems require cluster analysis to have new characteristics, that is, efficient scalability; It can process different attribute types in the data set, and find different clusters in the data set. The clustering is accurate and the data noise can be processed correctly. It performs well in high dimensional data sets. The clustering results have high reliability and so on.

The main steps of partitioning method are as follows. Firstly, it sets the partition k to be established. Partitions are created first with an initial partition and then using iterative relocation techniques to move objects from one group to another. To determine whether the result of partition is good or bad, the correlation degree of the objects in the cluster should be as high as possible and the difference between the clusters should be as large as possible. The traditional partition method can be extended to the subspace clustering, without traversing all the data space, reducing the amount of computation. In practical clustering applications, the most commonly used heuristic methods, such as k-mean method and k-center point algorithm, can gradually improve the quality of clustering to approximate the local optimal solution.

2.2 Paillier Homomorphic Encryption

Taking two large prime numbers p and q, set $n = pq$, and obtain $\varphi(p, q) = (p-1)(q-1)$. λ is defined by Carmichael function as the least common multiple of $(p-1)$ and $(q-1)$, that is, $\lambda = lcm(p-1, q-1)$, lcm means to take the least common multiple.

According to the definition of the $n-th$-order residual class puzzle, if an integer x is called the $n-th$-order residual class of module n^2, then there is an integer $y \in Z_{n^2}^*$, which makes $x = y^n modn^2$ valid. Where mod is the modulus value. $Z_{n^2}^* = Z_n \times Z_n^*$. Z_n is the set of all nonnegative integers. Z_n^* is the set of all numbers in set Z_n that satisfy $gcd(x, n) = 1$. gcd means to take the greatest common divisor.

Paillier homomorphic encryption [19] includes key generation, encryption and decryption.

1) Key generation. Randomly select two large prime numbers p, q and integer $y \in Z_{n^2}^*$, compute $n = pq$ and $\lambda = lcm(p-1, q-1)$, make $gcd[L(y^\lambda modn^2), n] = 1$, the public key is (n, g), and the private key is λ. After the key is generated, the random number $r \in Z_n$ is selected to encrypt the data, and the ciphertext c is obtained, while m is the encrypted information. The calculation is shown in Equation (1).

$$c = E(m, r) = y^m r^n modn^2. \tag{1}$$

2) Encryption. Based on the theory of compound residual hypothesis, the inverse operation of Equation (1) in the definition domain can be obtained by calculating λ from p and q. That is, for ciphertext c, the plaintext m can be obtained after being processed by Equation (2), where $L(i) = (i-1)/n$.

$$m = D(c, \lambda) = \frac{L(c^\lambda modn^2)}{L(y^\lambda modn^2)} modn. \tag{2}$$

3) Decryption. According to the addition homomorphism property of Paillier encryption algorithm, for the ciphertext $E(x)$ and $E(t)$ of data x and t, the relation shown in Equation (3) is satisfied.

$$\begin{aligned} D[E(x) \oplus E(t)modn^2] &= D[(y^x r_1^n) \oplus (y^t r_2^n)E(t)modn^2] \\ &= D[y^{x+t}(r_1 r_2)^n modn^2] \\ &= D[E(x+t)modn^2] \end{aligned} \tag{3}$$

By processing $E(x)$ and $E(t)$, $E(x+t)$ can be obtained, and the specific value of data $x+t$ also can be obtained. This principle can be extended to the case of multiple groups of data. The properties of Paillier algorithm provide flexible ideas for tamper-proof and security protection of data encryption.

3 The Proposed Ciphertext Encryption Algorithm

3.1 Encryption Algorithm

Secret key generation (Keygen). First, it chooses two strong prime numbers p and q. And with the properties of strong prime numbers, we can get two more prime numbers p' and q', where $p' = (p-1)/2$, $q' = (q-1)/2$. And then, assume $N = pq$, $\lambda = lcm(p-1, q-1)/2$. It selects a generation factor $g \in Z^*_{N^2}$ (where $Z^*_{N^2}$ is a set of non-zero integers less than N^2, $g = N+1$) to obtain the public key $p_k = N$ and the private key $s_k = \lambda$.

Encryption (Enc). Given a plaintext $m \in Z_N$ and a random number $r \in Z_N$, the ciphertext can be expressed as:

$$[m] = g^m r^N modN^2 = (1+mN)r^N modN^2.$$

Decryption (Dec). The private key is required to decrypt the ciphertext. First, it calculates:

$$[m]^\lambda modN^2 = (1+mN)^\lambda r^{\lambda N} modN^2 = 1 + m\lambda N.$$

Since $gcd(\lambda, N) = 1$, it can obtain the plaintext:

$$m = L([m]^\lambda modN^2)\lambda^{-1} modN.$$

Where $L(x) = (x-1)/N$, λ^{-1} satisfies $\lambda^{-1}\lambda \equiv 1modN$, and then it uses the residual theorem to find the value of λ^{-1}.

Key decomposition. It selects a parameter δ so that $\delta \equiv 0mod\lambda$ and $\delta \equiv 1modN^2$. Define a polynomial,

$$q(x) = \delta + \sum_{i=1}^{k-1} \beta_i x^i.$$

Where β_i is any number in $Z^*_{\lambda N^2}$, where $Z^*_{\lambda N^2}$ is a set of non-zero integers less than λN^2. Let $\alpha_1, \alpha_2, \cdots, \alpha_n \in Z^*_{\lambda N^2}$ be n different non-zero numbers. Set $s_k^{(i)} = q(\alpha_i)$ as part of the secret key and send it to part i.

Partial decryption (PDec). After receiving the ciphertext $[m]$, partial secret key $s_k^{(i)} = q(\alpha_i)$ is used to partially decrypt the ciphertext, and partial plaintext $T^{(i)}$ is obtained, namely,

$$T^i = [m]^{q(\alpha_i)} modN^2.$$

Merge decryption (TDec). Once $d(d \geq k)$ partially decrypted ciphertexts are received, let $S = T^{\tau_1}, T^{\tau_2}, \cdots, T^{\tau_d}$, the algorithm can select any k ciphertexts in the set S to decrypt it.

$$T'' = \prod_{l\in S}(T^{(l)})^{\Delta l, S(0)} modN^2.$$

Where $\Delta l, S(x) = \prod_{j\in S, j\neq l} \frac{x-\alpha_j}{\alpha_l-\alpha_j}$. So the plaintext m can be obtained by the following formula, namely,

$$m = L(T'').$$

Ciphertext refresh (CR). Once the ciphertext $[m]$ is received, CR algorithm can update the ciphertext without changing the plaintext. It selects a random number $r' \in Z_N$, and calculates,

$$[m]' = [m]r'^N = (rr')^N(1+mN)modN^2.$$

In addition, given $m \in Z_N$, there is

$$[m]^{N-1} = (1+(N-1)mN)r^{(N-1)N}modN^2 = [-m].$$

3.2 Ciphertext Computing Algorithm

In order to ensure that the data can be calculated in the case of encryption, and to operate the encrypted data in the case of protecting the private data and the secret key from being leaked, the following three ciphertext computing algorithms are proposed.

A. Ciphertext multiplication Algorithm (CTMA)

CTMA computes $[xy]$ securely when cloud storage provides two encrypted data $[x]$ and $[y]$ as inputs.

Step 1. Cloud storage selects two random numbers r_x, $r_y \in Z_N$, it calculates:

$$\begin{aligned} X &= [r_x][x] = [x+r_x]. \\ Y &= [r_y][x] = [y+r_y]. \\ X_1 &= P_{s_k^{(1)}}(X), Y_1 = P_{s_k^{(1)}}(Y). \end{aligned}$$

$P_{s_k^{(1)}}$ is the partial decryption (PDec) and, it sends X, Y, X_1 and Y_1 to the cloud computing center.

Step 2. The cloud computing center receives X, Y, X_1 and Y_1 and calculates:

$$T_x^{(i)} = P_{s_k^{(i)}}(X), T_y^{(i)} = P_{s_k^{(i)}}(Y).$$

Cloud computing center uses TDec to decrypt X and Y, get $x' = x + r_x$ and $y' = y + r_y$, then calculates $h = x'y'$. It uses p_k to encrypt h, that is, $H = [h]$, and sends H to cloud storage.

Step 3. Once H is received, cloud storage computes:

$$\begin{aligned} S_1 &= [x]^{N-r_y} = [-r_y x]. \\ S_2 &= [y]^{N-r_x} = [-r_x y]. \\ S_3 &= [r_y r_x]^{N-1} = [-r_y r_x]. \end{aligned}$$

Then, cloud storage computes $HS_1S_2S_3 = [(x+r_x)(y+r_y)-r_yx-r_xy-r_yr_x] = [x,y]$. Therefore, cloud storage and cloud computing centers can jointly compute $[xy]$.

B. Ciphertext comparison algorithm (CTCA)

Given two encrypted numbers $[x]$ and $[y]$, CTCA can be used to determine the relationship between the plaintexts of two encrypted data (i.e. $x > y$, $x < y$ or $x = y$).

Step 1. It selects two random numbers r, $l \in Z_N$ and calculates:

$$\begin{aligned} E &= [x]^r[y]^{N-r}[l] = [r(x-y)+l] \\ E_1 &= P_{s_k^{(1)}}(E). \end{aligned}$$

And it sends E, E_1 and $[l]$ to the cloud computing center.

Step 2. The cloud computing center receives E, E_1 and $[l]$ and computes:

$$T_e^{(i)} = P_{s_k^{(i)}}(E).$$

And it uses TDec to decrypt E to get $e = r(x-y)+l$. Then the cloud computing center compares e and l. If $e > l$, it denotes $x > y$, then 1 is sent to cloud storage; If $e = l$, it denotes $x = y$, then 0 is sent to the cloud storage. Otherwise $x < y$, -1 is sent to the cloud storage system.

Step 3. The cloud storage system receives results from the cloud computing system. If 1 is received, it means $x > y$; If 0 is received, then $x = y$; If -1 is received, then $x < y$.

C. Ciphertext logarithm algorithm (CTLA)

Given an encrypted number $[x]$, CTLA will calculate $[lnx]$ securely.

Step 1. It selects a random number $r_x \in Z_N$, and calculates $X = [x]^{r_x} = [xr_x]$, $R = [Inr_x]$. Since the cloud storage knows $s_k^{(1)}$, it can calculate:

$$T^{(1)} = P_{s_k^{(l)}}(X).$$

It sends X, R, and $T(l)$ to the cloud computing center.

Step 2. The cloud computing center receives X, R, and $T(l)$. It computes:

$$T^{(i)} = P_{s_k^{(i)}}(X).$$

Then it decrypts X with TDec and gets $x' = xr_x$ and then calculates:

$$L = [lnx'] = [lnx + lnr_x].$$

The cloud computing center sends L to the cloud storage.

Step 3. The cloud storage system receives L and calculates:

$$\begin{aligned} LR^{N-1} &= [lnx + lnr_x][lnr_x]^{N-1} \\ &= [ln(x)] \end{aligned}$$

So it can obtain $[lnx]$.

4 Experiment and Performance Analysis

In order to verify the reliability of this new scheme (HWKA) for English data encryption, HWKA and other similar algorithms are tested under the same conditions in the simulation environment, and the results are analyzed and compared. The experimental environment is shown in Table 1.

Table 1: Experiment environment

Name	Statement
Operating system	Win11
RAM	32GB
CPU	Interl(R) Core TMi6

In the experimental environment of Table 1, the HWKA in this paper is tested and verified, and it is compared with FHE [13], MKHE [3] and LPPA [6] under the same conditions. The time of encryption and decryption and the accuracy of data interaction are analyzed and compared.

In the process of English data encryption and decryption by Paillier algorithm, the size of n in the public key determines the complexity of the key, and also indirectly determines the encryption and decryption time. In this paper, keys with n values of 32 bit, 64 bit, 128 bit, 256 bit, 512 bit, 1024 bite and 2048 bit are generated. Multiple encryption and decryption experiments are conducted on randomly generated data of the same length, and the results are sorted and analyzed. The relationships between the execution time of encryption, secondary encryption and decryption and the key length of Paillier homomorphic encryption algorithm are obtained, as shown in Table 2 and Figure 1. The decryption time includes the decryption of the "secondary encryption" ciphertext and the decryption of the transaction data.

As can be seen from Figure 1, when the key length is below 1024 bit, the encryption and decryption time of Paillier homomorphic encryption algorithm increases slightly but it has little change with the increase of the key length. When the key length exceeds 1024 bit, the key is quite complicated, and the encryption and decryption time spent also increases proportionately, and the requirements on hardware also increase.

On the premise of encrypting and decrypting English data of different data lengths, the encryption and decryp-

Table 2: Relation between execution time of HWKA and key length/t

key length	First encryption time	Secondary encryption time	Decryption time
0	12.1	12.1	14.6
32	14.7	15.4	15.8
64	16.9	17.2	25.2
128	18.4	18.6	28.7
256	23.5	23.9	35.4
512	51.6	53.4	62.8
1024	130.5	151.8	191.7

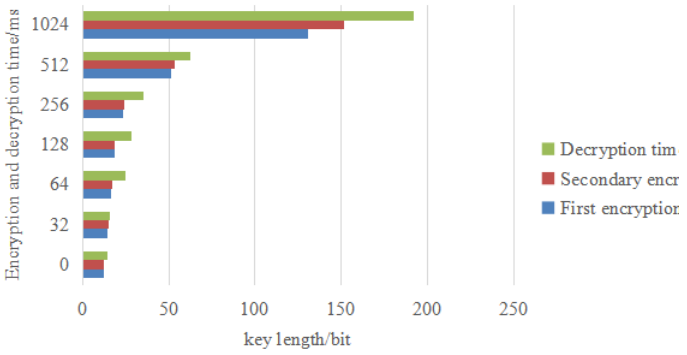


Figure 1: The visualization result of Table 2

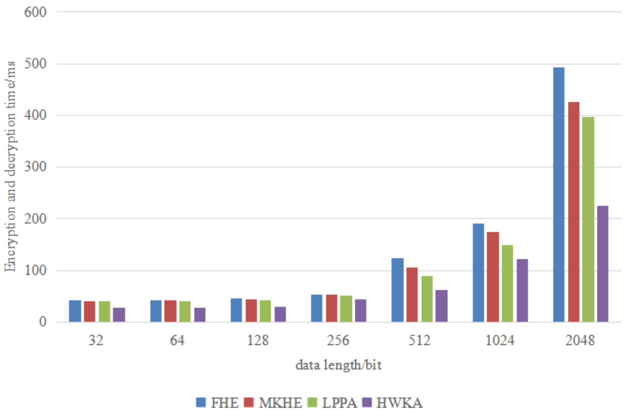


Figure 2: The visualization result of Table 3

tion execution time taken by the new method in this paper and other schemes is shown in Table 3 and Figure 2.

Table 3: Comparison of encryption and decryption execution time of different algorithms

data length	FHE	MKHE	LPPA	HWKA
32	41.6	40.7	39.8	27.6
64	42.7	41.8	40.9	28.3
128	44.9	43.7	42.6	29.8
256	53.8	52.4	52.1	44.7
512	123.7	105.2	89.7	61.8
1024	189.9	173.8	149.5	121.7
2048	492.7	426.5	397.2	224.5

Table 4: The accuracy of encryption and decryption with different algorithms/%

FHE	MKHE	LPPA	HWKA
90.8	92.7	95.3	98.6

As can be seen from Figure 2, compared with the encryption and decryption scheme of Paillier algorithm that only performs one-time encryption and decryption, the HWKA scheme slightly increases the execution time of encryption and decryption, but compared with other methods, the HWKA has certain advantages in the execution time of encryption and decryption process.

In addition to the efficiency of encryption and decryption, the success rate of English information encryption and decryption is also an important indicator to measure the data security scheme. The results are shown in Figure 3 and Table 4.

As can be seen from Figure 3, compared with other algorithms, under the same key length and the same encrypted information, the encryption and decryption accuracy rate of the proposed algorithm reaches 98.6%. This shows that the scheme has a positive effect on ensuring the reliability of data transmission in power trading process.

5 Conclusions

In this paper, we improve the clustering method by changing the clustering process under the condition of vertical distribution of data, and introduce homomorphic encryption technology in the clustering. Then this method is applied to the k-center clustering algorithm, which makes the security of data further guaranteed. In this paper, the complexity of communication and accuracy of the new algorithm are analyzed. In the last experiment, it is proved

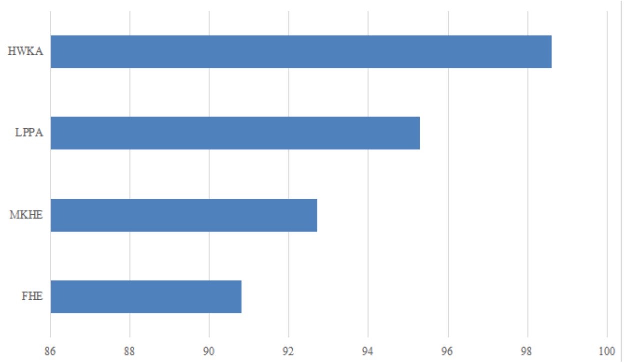


Figure 3: The accuracy of encryption and decryption with different algorithms

that the algorithm in this paper has better performance compared with other algorithms of the same type, and the communication encryption and decryption cost has been greatly reduced, which makes the computing load in distributed mode more balanced. On the other hand, this paper also studies and analyzes the privacy security of data mining, and proves the role of order preserving encryption and homomorphic encryption in the privacy security of clustering algorithm.

References

[1] S. Akleylek, K. Seyhan, "A probably secure bi-GISIS based modified AKE scheme with reusable keys," *IEEE Access*, vol. 8, pp. 26210-26222, 2020.

[2] R. Aversa, P. Coronica, C. De Nobili, *et al.*, "Deep learning, feature learning, and clustering analysis for sem image classification," *Data Intelligence*, vol. 2, no. 4, pp. 513-528, 2020.

[3] V. N. R. Bandaru, P. Visalakshi, "Block chain enabled auditing with optimal multi-key homomorphic encryption technique for public cloud computing environment," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 22, pp. e7128, 2022.

[4] Z. Chen, Z. Liu, L. Wang, "A modified model predictive control method for frequency regulation of microgrids under status feedback attacks and time-delay attacks," *International Journal of Electrical Power & Energy Systems*, vol. 137, pp. 107713, 2022.

[5] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments", *International Journal of Network Security*, vol. 16, no. 1, pp. 1-13, 2014.

[6] C. Guo, X. Jiang, K. K. R. Choo, *et al.* , "Lightweight privacy preserving data aggregation with batch verification for smart grid," *Future Generation Computer Systems*, vol. 112, pp. 512-523, 2020.

[7] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 5, 2017.

[8] H. Kim, S. H. Kim, J. Y. Hwang, *et al.*, "Efficient privacy-preserving machine learning for blockchain network," *IEEE Access*, vol. 7, pp. 136481-136495, 2019.

[9] C. W. Liu, W. F. Hsien, C. C. Yang, M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage", *International Journal of Network Security*, vol. 18, no. 5, pp. 900-916, 2016.

[10] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for shared data storage with user revocation in cloud computing", *International Journal of Network Security*, vol. 18, no. 4, pp. 650–666, 2016.

[11] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *International Journal of Network Security*, vol. 18, no. 5, pp. 900–916, 2016.

[12] J. Loftus, A. May, N. P. Smart, *et al.*, "On cca-secure fully homomorphic encryption," *Cryptology ePrint Archive*, 2010.

[13] M. A. Mohammed, F. S. Abed, "Cloud storage protection scheme based on fully homomorphic encryption," *ARO-The Scientific Journal of Koya University*, vol. 8, no. 2, pp. 40-47, 2020.

[14] S. Nandi, S. Krishnaswamy, B. Zolfaghari, *et al.* , "Key-dependent feedback configuration matrix of primitive ρ–LFSR and resistance to some known plaintext attacks," *IEEE Access*, vol. 10, pp. 44840-44854, 2022.

[15] H. Orii, K. Hatano, H. Tanaka, *et al.*, "An image conversion method for color discriminability compensation of colorblindness using CycleGAN," *IEICE Proceedings Series*, vol. 69(RS2-6), 2022.

[16] A. K. Sandhu, "Big data with cloud computing: Discussions and challenges," *Big Data Mining and Analytics*, vol. 5, no. 1, pp. 32-40, 2021.

[17] X. Wang, S. Yin, M. Shafiq, *et al.*, "A new V-net convolutional neural network based on four-dimensional hyperchaotic system for medical image encryption," *Security and Communication Networks*, vol. 2022, pp. 1-14, 2022.

[18] A. Wood, K. Najarian, D. Kahrobaei, "Homomorphic encryption for machine learning in medicine and bioinformatics," *ACM Computing Surveys*, vol. 53, no. 4, pp. 1-35, 2020.

[19] S. Yin, H. Li, L. Teng, "A novel proxy re-encryption scheme based on identity property and stateless broadcast encryption under cloud environment," *International Journal of Network Security*, vol. 21, no. 5, pp. 797-803, 2019.

[20] M. Zhang, S. Huang, G. Shen, *et al.*, "PPNNP: A privacy-preserving neural network prediction with separated data providers using multi-client inner-product encryption," *Computer Standards & Interfaces*, vol. 84, pp. 103678, 2023.

[21] H. Zheng, Z. Gan, X. Li, *et al.*, “A green neural network with privacy preservation and interpretability,” in *IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pp. 606-614, 2022.

Biography

Haiying Liu biography. Haiying Liu is with School of Foreign Languages, Zhengzhou University of Science and Technology, born in Nanyang, Henan Province, 450064, China. Research interests: English data analysis.

Tackling Pakistan's Cyber Security Challenges: A Comprehensive Approach

Muhammad Ibrar[1], Hang Li[1], Jiachi Wang[1], and Shahid Karim[2]
(Corresponding author: Hang Li)

Software College, Shenyang Normal University, Shenyang, China[1]
Shenyang 110034 China
Research & Development Institute of Northwestern Polytechnical University in Shenzhen[2]
Shenzhen 518057, China
Email: lihangsoft@163.com

(Received Dec. 26, 2022; Revised and Accepted Apr. 16, 2023; First Online Apr. 30, 2023)
The Special Issue on Computational Intelligence Networks for Privacy and Security in Evolving Internet of Multimedia Things
Special Editor: Prof. Shoulin Yin (Harbin Institute of Technology)

Abstract

The rapid growth of internet usage and its various applications, such as banking, shopping, social networks, and other business-related activities, has moved people closer to cyberspace. The implications of this growth have also come to Pakistan, where cyber security and threats are becoming an essential aspect of cyberspace. This paper will discuss the challenges Pakistan faces regarding cyber security and the steps that need to be taken to address those threats. It will focus on the recent cyberattacks, such as ransomware and distributed denial-of-service (DDoS), which have disrupted vital services in Pakistan. Furthermore, it will discuss the potential solutions and strategies that can be used to ensure that all users in Pakistan have a secure experience when accessing the internet. Finally, it will also present recommendations for organizations to consider when developing cyber security policies for their operations in Pakistan.

Keywords: Challenges for Pakistan; Cyber Security; Cyber Security Policies; Cyberspace

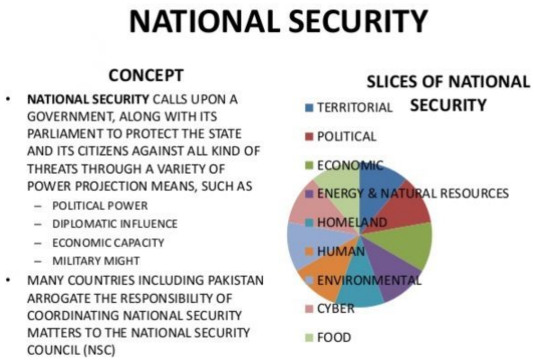
1 Introduction

Cyber Security is the practice of protecting systems, networks, and programs from digital attacks. These attacks usually aim to access, change, or destroy sensitive information, extort money from users, or interrupt normal business processes [15]. Cyber security involves the prevention of, detection of, and response to security incidents that take place in a digital environment [22, 23]. It is a rapidly evolving field that is becoming increasingly important today [5,24]. The amount of data created and stored online increases each year. This growing quantity of data increases the risk of cyber-attacks as hackers look for opportunities to exploit it. Organizations must put in place various measures to protect their confidential information and prevent malicious actors from gaining unauthorized access or disrupting their operations [17].

Pakistan is one of the developing countries where cyber security is yet to gain momentum due to a lack of awareness, expertise, and resources available for this purpose. The country faces challenges ranging from a lack of technical capacity to develop and implement effective cyber security strategies, a lack of legal framework, an absence of public-private partnerships, and funding constraints [4]. In addition, Pakistan also faces several social challenges, such as low levels of education, poverty, and a lack of access to technology, making it difficult for people to protect themselves from cyber threats. Furthermore, due to its proximity to volatile countries such as Afghanistan and Iran, Pakistan is also at risk of becoming a target for cyber-attacks by state-sponsored hackers [7]. The government has taken several steps to address these challenges, such as establishing the Cyber Emergency Response Team (CERT), which monitors the country's IT networks and responds to any cyber threats it detects. The government also provides awareness campaigns to educate people about cyber security threats and how to protect themselves online as shown in Figure 1.

However, these efforts are just a drop in the ocean, considering the scale of the problem. The country needs to develop more comprehensive policies and strategies on cyber security to combat the threat posed by cybercriminals effectively [8].

With the world increasingly becoming interconnected through technology, the threat of cybercrime is also on the rise. As a result, countries worldwide are making great strides to increase their cyber security measures to prevent cyberattacks and data breaches. Pakistan is no exception. The government is constantly grappling with



NATIONAL SECURITY

CONCEPT

- NATIONAL SECURITY CALLS UPON A GOVERNMENT, ALONG WITH ITS PARLIAMENT TO PROTECT THE STATE AND ITS CITIZENS AGAINST ALL KIND OF THREATS THROUGH A VARIETY OF POWER PROJECTION MEANS, SUCH AS
  - POLITICAL POWER
  - DIPLOMATIC INFLUENCE
  - ECONOMIC CAPACITY
  - MILITARY MIGHT
- MANY COUNTRIES INCLUDING PAKISTAN ARROGATE THE RESPONSIBILITY OF COORDINATING NATIONAL SECURITY MATTERS TO THE NATIONAL SECURITY COUNCIL (NSC)

SLICES OF NATIONAL SECURITY

Figure 1: The theoretical concept of national security

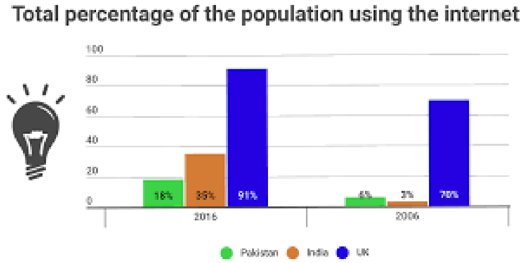


Figure 2: Population using the internet in Pakistan, India, and UK

various cybersecurity challenges, from the risk of malicious actors exploiting its networks and infrastructure to the need to update critical information systems and protect citizens data [15].

The percentage of the population using the internet in Pakistan, India, and the UK is shown in Figure 2. In recent years, Pakistan has made some critical reforms to improve its cyber security. In 2019, the Ministry of Information Technology and Telecommunications launched the National Response Centre for Cyber Crimes (NR3C). This center has been instrumental in increasing awareness about cyber threats and coordinating efforts between law enforcement and other relevant agencies on cybercrime investigations. The center also develops training materials for law enforcement officers and raises public awareness about cyber safety and security best practices [9].

The government has also enacted several laws to help protect against cybercrime. In August 2020, the National Assembly passed the Prevention of Electronic Crimes Act (PECA) 2020, which criminalizes various activities, including hacking, fraud, intellectual property rights infringement, terrorism, extortion, and identity theft. The government has also issued guidelines for protecting the personal data of individuals, including measures to prevent unauthorized access or disclosure [10]. Despite these efforts, several challenges remain that need to be addressed to ensure a secure cyberspace in Pakistan for example, adequate resources and personnel to be improved to investigate cyber threats and enforce cyber laws. In addition, there are areas for improvement in existing legal frameworks that make it challenging to pursue sophisticated cybercrime cases. Furthermore, the country's critical infrastructure remains vulnerable to attack due to outdated technology and weak security measures. Finally, there is a need to develop comprehensive policies and strategies that address both domestic and international threats [11].

Overall, it is clear that Pakistan still faces a range of cyber security challenges. However, by taking steps such as improving laws and regulations, strengthening infrastructure protection measures, and enhancing collaboration between relevant stakeholders, the country can make progress toward achieving a safe and secure cyberspace for its citizens.

2 Research Question

What strategies can be used to improve cyber security in Pakistan and address the challenges faced by the country in this regard?

3 Research Objectives

1) To measure the levels of cyber security in Pakistan and assess the current efforts to mitigate the risk of cyber-attacks.

2) To identify and analyze the critical challenge of cyber security in Pakistan and suggest ways to strengthen the existing protection infrastructure.

3) To assess the extent to which Pakistani businesses are vulnerable to cyber threats and suggest measures for their protection.

4) To investigate the government's and other stakeholders' role in promoting cyber security in the country.

5) To understand the various tools and technologies hackers use to penetrate Pakistani networks and devise practical solutions to combat them.

6) To evaluate the preparedness of Pakistan to counter emerging threats posed by cybercriminals, such as phishing and malware attacks.

7) To examine the efficacy of existing international laws, regulations, and practices in safeguarding Pakistani networks from external cyber threats.

8) To develop an effective strategy for raising public awareness of cyber security in Pakistan and suggest necessary steps for its implementation.

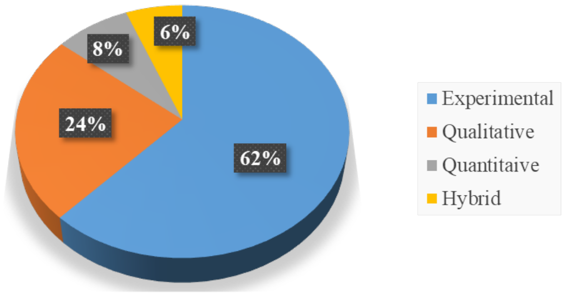


Figure 3: Distribution of articles according to methodology

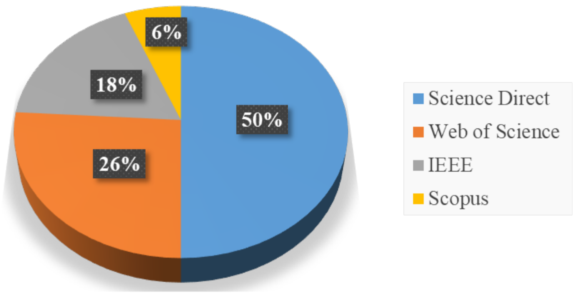


Figure 4: Distribution of articles according to the percentage of databases

4 Research Problem

Cyber security is an increasing concern in Pakistan. The country has faced numerous cyber-attacks, from traditional cybercrime and identity theft to political attacks targeting government and private institutions. Despite the growing cyber security threats, there needs to be more knowledge, awareness, and preparedness among the public and organizations to counter such threats. Moreover, the country needs more adequate laws and infrastructure to combat cybercrime effectively. As such, it is essential to research the cyber security challenges faced by Pakistan and develop strategies to safeguard its citizens and institutions from these threats. This research will focus on identifying the major cyber security threats faced by Pakistan, understanding their impact on the country, analyzing the current strategies adopted by the government to address these challenges, and proposing potential solutions to these cyber security issues.

5 Literature Review

Cyber security is a significant concern today, and Pakistan is no exception. As technology advances, the security of digital systems and networks has become increasingly important in maintaining the integrity of nations and businesses. The prior research for cyber security has been categorized based on research methodology (i.e., experimental, qualitative, quantitative, and hybrid), as shown in Figure 3. Figure 4 displays the percentage of databases containing the number of papers published for cyber security. We have selected four well-known databases to evaluate the statistics for this review. The experience of Pakistan in this regard has been mixed, with some successes and many challenges. This paper will review the existing literature on cyber security in Pakistan, examining the country's main challenges and possible solutions to address these issues [12].

It is essential to understand the scope of the cyber security problem in Pakistan. Over the past few years, several documented cases of cyberattacks originated in Pakistan, and hackers are targeting an increasing number of individuals. In addition, state actors are involved in state-sponsored cyber espionage and other malicious activities. These threats come from both external actors as well as domestic criminals. Furthermore, due to lax regulations on data protection, businesses are more vulnerable to data breaches and cybercrime in general [13]. One of the main challenges faced by Pakistan is the need for a practical legal framework governing cyber security issues. Several laws and regulations exist on the books, but they must be more comprehensive to address all aspects of cyber security. This system has hindered efforts by public and private organizations to address cyber threats with any degree of permanence. Furthermore, enforcement of these laws could be more robust due to a lack of adequate resources or political will [14]. According to [3] modern threats and types of data expected to be attacked are presented in Table 1.

Another critical challenge is more awareness and education about cyber security among citizens and government agencies. People have become increasingly dependent on technology without understanding its associated risks, leading to a higher incidence of online fraud, identity theft, and other malicious activities. Similarly, governmental institutions lack sufficient resources and training to address the challenges cyber criminals pose effectively. Fourth, another major challenge for Pakistan is the need for skilled professionals to protect networks from attackers. While some professionals are currently working in the field, their numbers are insufficient to address the growing problem that Pakistan faces from external actors and domestic criminals [16]. Additionally, there are also numerous challenges related to international cooperation on cyber security issues faced by Pakistan. For example, some countries are unwilling to cooperate with Pakistan due to sensitive information being shared across borders or because they may seek a competitive advantage over Pakistan regarding cyberspace capabilities. Additionally, government agencies in other countries may be unwilling to share information or provide assistance due to security concerns or because they view Pakistan as a potential source of cybercrime or terrorism [18]. While there have been some successes in addressing cyber secu-

Table 1: Modern threats and types of data expected to be attacked

Threats	Domains
Surveillance	Social, E-commerce, environment, and political governance
User Profile	Actives and behavioral characteristics
Cyberstalking	Harassment and intimidation
Clickjacking	Press the link or like button, move cursors, use the camera and microphone
Location Privacy	Geotagging
Identity profile cloning	Creating a fake profile
Information Leakage	Health, infrastructure, operational, and intellectual property information
Fake profile Attacks	User information
De-anonymization	Health services, social media, and E-commerce trades
Inference Attacks	Prediction Sensitive, political, religious, and educational information

rity threats in Pakistan, numerous challenges still need to be addressed before meaningful progress can be made. These include a lack of adequate legal frameworks and enforcement mechanisms; inadequate public awareness; insufficient numbers of skilled professionals; and difficulty with international cooperation on cyber security issues. With improved governance and better regulations, Pakistani citizens and businesses can be better protected from cyber threats.

6 Theoretical Framework

Cyber security is the collective set of activities intended to protect computers, networks, programs, and data from unauthorized access, exploitation, and disruption. Pakistan is no stranger to cyber threats. As a rapidly developing economy, the country faces various cybersecurity challenges due to its limited capacity to handle sophisticated cyber-security threats, weak policy frameworks, and limited access to technology and know-how for proactive security measures. The cyber security challenge faced by Pakistan is multi-faceted, as it involves both state and non-state actors who use the internet for criminal activities, espionage, hacking, and manipulation of information. In particular, the country has been subject to frequent cyber-attacks from state actors in the region and abroad. Furthermore, Pakistan faces challenges from cybercrime, ranging from fraud and hacking to identity theft and phishing. As more organizations increasingly connect to the internet and rely on digital systems for their operations, they face an increased risk of becoming cyber-attack targets [19]. To address these challenges, effective policy measures that promote cyber security amongst organizations operating in Pakistan need to be established. There is a need to develop a comprehensive regulatory framework that will ensure the safety of digital systems and networks. Furthermore, organizations must invest in effective cyber security measures such as firewalls and encryption technologies to protect their systems from threats. Additionally, organizations should adopt effective incident response plans that will enable them to address any incidents occurring due to cyber security threats quickly [20].

Organizations must invest in capacity-building initiatives to develop adequate cybersecurity capabilities and provide technical assistance for properly implementing security measures. Additionally, Pakistan must work closely with regional and global partners on international forums such as the United Nations Group of Governmental Experts on Cyber Security (UNGGE) to coordinate better efforts to enhance its cyber security capabilities [21]. At a national level, initiatives such as awareness campaigns must be undertaken to educate individuals on their role in maintaining existing levels of cyber security. Such efforts can improve Pakistan's security by encouraging citizens and organizations to adopt proactive measures against threats.

7 Cyber Design Structure

1) Understand the fundamentals of user experience design: User experience (UX) design is a process for understanding how users interact with digital products and services to improve usability, satisfaction, and efficiency. It includes researching user needs designing and visual elements such as visuals, animations, illustrations, and other graphical elements that enhance the user's experience.

2) Develop a security-first mindset: Security is integral to cyber design since it helps protect users from malicious threats or data theft. By creating secure designs and processes from the ground up, you can help ensure that your product or service meets industry best practices for keeping users safe online.

3) Establish innovative-cation protocols: Authentication protocols are vital when protecting sensitive data stored on a system or website. An exemplary cation protocol should combine factors such as passwords, biometrics, two-factor authentication (2FA), URL scanning checks, and captcha challenges to bust

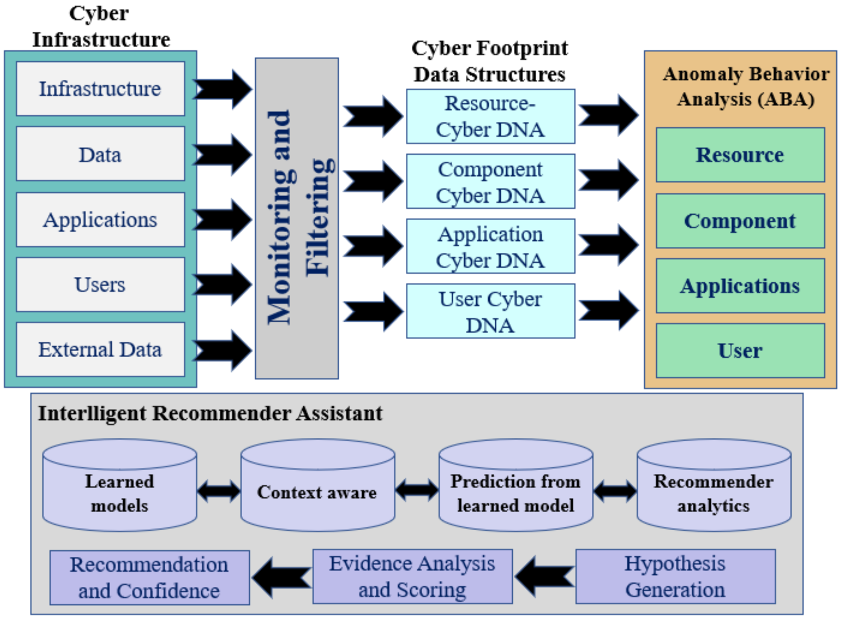


Figure 5: Intelligent Cyber security assistant architecture ***The figure did not cited in this text.***

layers of protection against unauthorized attempts by hackers or malicious actors.

4) Keep performance in mind: Performance is critical when providing a great user experience on any digital platform; poor performance leads to poor usability, which can n reduce customer satisfaction levels over time if not addressed quickly enough. It is crucial essential performance in mind while designing applications as this will ensure that they remain responsive even under heavy load times or network delays due to peak traffic periods or the geographic location of customers using your product/service etc.

5) Utilize analytics tools for testing & optimization: Analytics tools are essential for measuring how well users interact with your product/service over time; these insights can then be used to improve design decisions based upon customer feedback, helping shape future releases towards better customer experiences overall!

8 Cyber security and Pakistan

Pakistan is vulnerable to cyber-attacks as its infrastructure and security systems need more resources, personnel, and technology resources against cyber-attacks, making it an attractive target for hackers. Pakistan's economy also suffers from the lack of enforcement of laws related to cyber security. Cybercrime legislation is badly needed in Pakistan, but the government has yet to pass any significant laws that would better protect citizens against attacks. In addition, numerous cybercrimes such as fraud, identity theft, illegal downloads, and hacking remain unresolved due to a lack of resources dedicated to fighting these crimes [25].

For Pakistan to be better protected from cyber threats, it must strengthen its legal framework on cyber security and increase investment into technological solutions that can defend networks from attack. The government needs more effective coordination between law enforcement agencies and tech companies to identify malicious actors online and take appropriate action against them. Additionally, organizations need to promote a more robust culture of cyber security by providing proper training on security protocols so that potential breaches can be identified before they cause severe damage or loss. Furthermore, public campaigns should be conducted nationwide to educate users about what actions to take when using the internet to minimize their risk of exposure to cyber threats [1].

Pakistan's cyber security is a significant concern due to its weak infrastructure, limited resources, and lack of expertise. The Cyber preparedness, elements, and strategic areas are shown in Figure 6. The country is vulnerable to various cyber threats, including data theft or manipulation, ransomware attacks, website defacement, distributed denial-of-service (DDoS) attacks, keylogging, and phishing. Pakistan's legal framework for cybercrime prevention and response is also inadequate due to its lack of expertise in this area. In addition, there is a lack of public awareness of the importance of cyber security and limited resources allocated for developing and implementing effective cyber security measures. Furthermore, government agencies have failed to provide contractors with relevant security training so that they can adequately protect sensitive information stored within their systems. These vulnerabilities are further compounded by the fact that

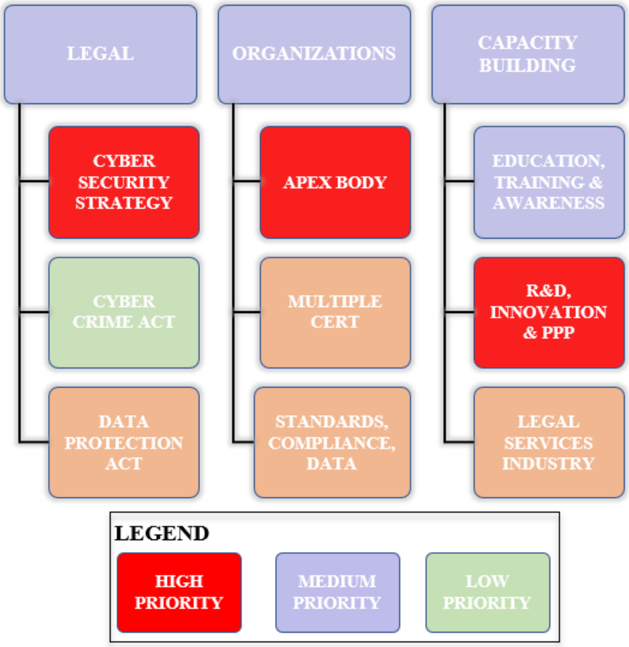


Figure 6: Cyber preparedness, strategic trust areas, and elements

Pakistan has one of the lowest internet penetration rates in Asia, with only 37 million internet users out of its estimated population of more than 200 million. This low level makes it difficult for citizens to access vital online services such as banking or health care, which puts them at risk of being targeted by malicious actors who could use these weaknesses to gain access to personal data or inflict damage upon individuals or organizations operating within Pakistan's borders [6].

9 Pakistan's Preparedness for Cyber Threats

Pakistan is actively engaging in initiatives to protect itself from cyber threats. The government has established the National Cyber Security Authority (NCSA). It is developing comprehensive strategies, policies, and procedures for protecting the country from cyber-attacks [12]. The NCSA also works with various stakeholders, including industry groups, universities, law enforcement agencies, and even other countries, to develop a holistic approach to tackling the issue of cyber security in Pakistan as shown in Table 2. It has put together an extensive policy framework that guides how to deal with incidents of cybercrime as well as directives on how the public sector should respond. Moreover, law enforcement agencies like the Federal Investigation Agency (FIA), the Ministry of Interior's National Response Centre for Cybercrime (NR3C), and the Ministry of Information Technology & Telecommunications' Cyber Crime Reporting Portal are also playing their part in providing quick response to incidents related to cyber security [2].

To ensure an effective response against malicious online activities, Pakistan Telecommunication Authority (PTA) has blocked more than 800 websites hosting anti-state content since 2016. It has taken further proactive steps like setting up a 24/7 monitoring system for social media websites & networks [20].

The last cyber security ranking is shown in Table 2 for the comparison of different countries. In addition, various educational programs are being conducted throughout the country regarding understanding best practices when dealing with cyber security issues so that people become aware of these issues, which can help reduce risks associated with them in the future.

10 Conclusion and Recommendations

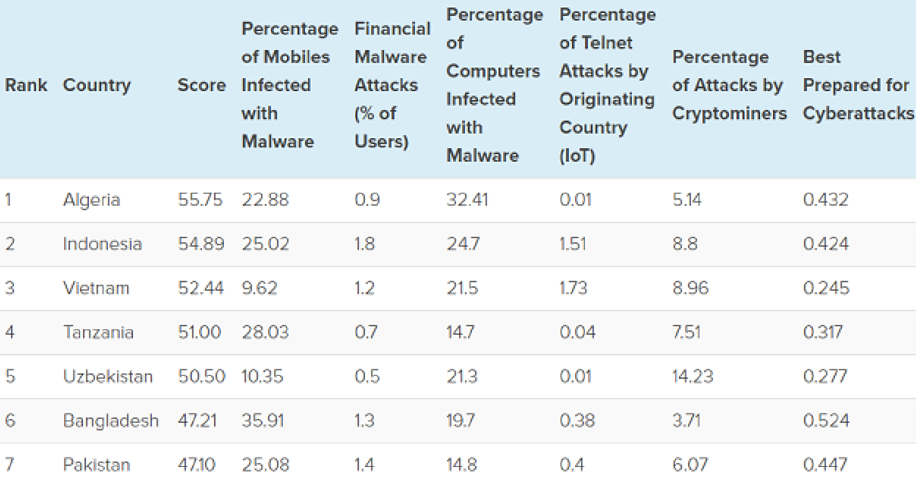
In conclusion, Pakistan's current cyber security state is inadequate and poses a severe risk to national security. Numerous reports have been of government and private network attacks, often with devastating consequences. The National Assembly of Pakistan has passed the Cyber Security Act, 2018, which aims to protect citizens from cybercrime by strengthening the legal framework and providing safeguards against cybercrimes. However, the effectiveness of the law is yet to be seen. The lack of competent personnel and resources within governmental agencies means that many organizations must be equipped to respond effectively to threats or defend against them. Additionally, public awareness about cyber security measures must be raised for citizens and businesses to protect themselves from malicious actors. By developing a culture of preparing for and protecting against future attacks, we can reduce the number and severity of incidents our society suffers in this digital age.

1) Improve Governance & Legislation: The government should create and implement strict cyber security laws and regulations to protect the country's cyberspace from malicious activities. This process should include watching citizens' data, imposing penalties on cybercriminals, and protecting critical infrastructure such as power grids and banking systems.

2) Focus on Network Security: Companies need to increase their security investments and focus on updating their networks with the latest cybersecurity solutions. This strategy will enable them to detect malicious activity quickly and prevent significant data breaches.

3) Train Cyber Security Professionals: Companies and government agencies should focus on training cyber security professionals to deal with the ever-growing number of cyber security threats they face.

Table 2: Countries ranking concerning cyber security



Rank	Country	Score	Percentage of Mobiles Infected with Malware	Financial Malware Attacks (% of Users)	Percentage of Computers Infected with Malware	Percentage of Telnet Attacks by Originating Country (IoT)	Percentage of Attacks by Cryptominers	Best Prepared for Cyberattacks
1	Algeria	55.75	22.88	0.9	32.41	0.01	5.14	0.432
2	Indonesia	54.89	25.02	1.8	24.7	1.51	8.8	0.424
3	Vietnam	52.44	9.62	1.2	21.5	1.73	8.96	0.245
4	Tanzania	51.00	28.03	0.7	14.7	0.04	7.51	0.317
5	Uzbekistan	50.50	10.35	0.5	21.3	0.01	14.23	0.277
6	Bangladesh	47.21	35.91	1.3	19.7	0.38	3.71	0.524
7	Pakistan	47.10	25.08	1.4	14.8	0.4	6.07	0.447

4) Increase Cyber Security Awareness: The government and companies should increase general cyber security awareness among the citizens by running campaigns in schools, universities, and other public places.

5) Collaborate with Cybersecurity Companies: The Pakistani government should collaborate with private cybersecurity companies to develop cutting-edge solutions for protecting cyberspace from malicious activities.

References

[1] S. AlDaajeh, H. Saleous, S. Alrabaee, *et al.*, "The role of national cybersecurity strategies on the improvement of cybersecurity education," *Computers & Security*, vol. 119, pp. 102754, 2022.

[2] A. Ali, A. W. Septyanto, I. Chaudhary, *et al.*, "Applied artificial intelligence as event horizon of cyber security," in *International Conference on Business Analytics for Technology and Security (ICBATS'22)*, IEEE, pp. 1-7, 2022.

[3] H. Almarabeh, A. Sulieman, "The impact of cyber threats on social networking sites," *International Journal of Advanced Research in Computer Science*, vol. 10, no. 2, 2019.

[4] J. H. Awan, S. Memon, M. H. Shah, *et al.*, "Security of eGovernment services and challenges in Pakistan," in *SAI Computing Conference (SAI'16)*, IEEE, pp. 1082-1085, 2016.

[5] E. W. Baker, "A model for the impact of cybersecurity infrastructure on economic development in emerging economies: evaluating the contrasting cases of India and Pakistan," *Information Technology for Development*, vol. 20, no. 2, pp. 122-139, 2014.

[6] A. Corallo, M. Lazoi, M. Lezzi, *et al.*, "Cybersecurity awareness in the context of the Industrial Internet of Things: A systematic literature review," *Computers in Industry*, vol. 137, pp. 103614, 2022.

[7] S. Farid, M. Alam, G. Qaiser, A. Ul Haq, J. A. Itmazi, "Security threats and measures in E-learning in Pakistan: A review," *Technical Journal, University of Engineering and Technology (UET) Taxila, Pakistan*, vol. 22, no. 3, pp. 98-107, 2017.

[8] A. Farooq, S. R. U. Kakakhel, "Information security awareness: Comparing perceptions and training preferences," in *2nd National Conference on Information Assurance (NCIA'13)*, IEEE, pp. 53-57, 2013.

[9] M. A. Firdous, "Formulation of Pakistan's cyber security policy," *CISS Insight Journal*, vol. 6, no. 1, pp. 70-94, 2018.

[10] Z. Hussain, D. Das, Z. A. Bhutto, *et al.*, "E-banking challenges in Pakistan: an empirical study," *Journal of Computer and Communications*, vol. 5, no. 2, pp. 1-6, 2017.

[11] A. Khan, M. Ibrahim, A. Hussain, "An exploratory prioritization of factors affecting current state of information security in Pakistani university libraries," *International Journal of Information Management Data Insights*, vol. 1, no. 2, pp. 100015, 2021.

[12] M. F. Khan, A. Raza, N. Naseer, "Cyber security and challenges faced by Pakistan," *Pakistan Journal of International Affairs*, vol. 4, no. 4, 2021.

[13] M. I. Khan, "Cyber-warfare: Implications for the national security of Pakistan," *NDU Journal*, vol. 117-132, 2019.

[14] S. Khattak, S. Jan, I. Ahmad, *et al.*, "An effective security assessment approach for Internet banking services via deep analysis of multimedia data," *Multimedia Systems*, vol. 27, pp. 733-751, 2021.

[15] M. Lyytikinen, P. Yadav, A. T. R. Wibben, *et al.*, "Unruly wives in the household: Toward feminist genealogies for peace research," *Cooperation and Conflict*, vol. 56, no. 1, pp. 3-25, 2021.

[16] Z. U. A. Malik, H. M. Xing, S. Malik, *et al.*, "Cyber security situation in Pakistan: A critical analysis," *PalArch's Journal of Archaeology of Egypt/Egyptology*, vol. 19, no. 1, pp. 23-32, 2022.

[17] A. Naha, "Emerging cyber security threats: India's concerns and options," *International Journal of Politics and Security*, vol. 4, no. 1, pp. 170-200, 2022.

[18] D. R. Naseer, D. M. Amin, "Cyber-threats to strategic networks: Challenges for Pakistan's security," *South Asian Studies*, vol. 33, no. 1, 2020.

[19] S. Rasool, "Cyber security threat in Pakistan: Causes, challenges and way forward," *International Scientific Online Journal*, vol. 12, pp. 21-34, 2015.

[20] K. Shaukat, T. M. Alam, I. A. Hameed, *et al.*, "A review on security challenges in internet of things (IoT)," in *26th International Conference on Automation and Computing (ICAC'21)*, IEEE, pp. 1-6, 2021.

[21] K. Shaukat, S. Luo, V. Varadharajan, *et al.*, "Performance comparison and current challenges of using machine learning techniques in cybersecurity," *Energies*, vol. 13, no. 10, pp. 2509, 2020.

[22] J. R. Sun, M. S. Hwang, "A new investigation approach for tracing source IP in DDoS attack from proxy server", in *Intelligent Systems and Applications*, pp. 850-857, 2015.

[23] J. R. Sun, M. L. Shih, M. S. Hwang, "A survey of digital evidences forensic and cybercrime investigation procedure", *International Journal of Network Security*, vol. 17, no. 5, pp. 497-509, 2015.

[24] J. R. Sun, M. L. Shih, M. S. Hwang, "Cases study and analysis of the court judgement of cybercrimes in Taiwan", *International Journal of Law, Crime and Justice*, vol. 43, no. 4, pp. 412-423, 2015.

[25] F. Z. Syed, S. Javed, "Deterrence: A security strategy against non traditional security threats to Pakistan," *International Journal of Social Sciences and Management*, vol. 4, no. 4, pp. 267-274, 2017.

Biography

Muhammad Ibrar biography. Muhammad Ibrar is with the Software College, Shenyang Normal University. His major is computer science, information secure.

Hang Li biography. Prof. Hang Li is with the Software College, Shenyang Normal University. His major is computer science, image processing.

Jiachi Wang biography. Jiachi Wang is with the Software College, Shenyang Normal University. His major is computer science, image processing.

Shahid Karim biography. Shahid Karim is with the Research & Development Institute of Northwestern Polytechnical University in Shenzhen.

Research on Local Differential Privacy Protection of High Dimensional Data in Embedded System Based on Hybrid Differential Swarm Algorithm

Shiru Sun[1], Bin Wang[2], and Hongwei Zhang[3]
(Corresponding author: Shiru Sun)

School of Electronic and Electrical Engineering, Zhengzhou University of Science and Technology[1]
College of Big Data and Artificial Intelligence, Zhengzhou University of Science and Technology[2]
College of Civil Engineering and Architecture, Zhengzhou University of Science and Technology[3]
Zhengzhou 450064, China
Email: xdwangxd@163.com

(Received Dec. 29, 2022; Revised and Accepted Apr. 16, 2023; First Online Apr. 30, 2023)

The Special Issue on Computational Intelligence Networks for Privacy and Security in Evolving Internet of Multimedia Things
Special Editor: Prof. Shoulin Yin (Harbin Institute of Technology)

Abstract

Deep neural networks have significant gradient redundancy in gradient descent. Therefore, excessive noise is introduced when the differential privacy mechanism is used to resist member inference attacks. This paper proposes a local differential privacy protection theme of high dimensional data in embedded systems based on a hybrid differential swarm algorithm to solve this problem. Firstly, the data source differential privacy protection algorithm is used to disturb the client data set and generate the disturbed data set to protect the privacy of the original local data set. Then, a hybrid differential swarm algorithm reduces the high-dimensional data set to multiple low-dimensional attribute sets, and a new data set is synthesized. The algorithm uses the maximum spanning tree criterion to get the initial population. Then the crossover and mutation rules in differential evolution algorithms optimize the initial population. Using the differential evolution algorithm, the swarm algorithm is applied to the mutation stage, the optimization and improvement crossover stage, respectively, and the adaptive cloud theory is applied to the selection stage to select the generated individuals. Finally, the proposed algorithm is verified on the standard data sets MNIST and CIFAR-10 to more effectively bridge the gap between the proposed algorithm and the non-privacy model.

Keywords: Embedded System; High Dimensional Data; Hybrid Differential Swarm Algorithm; Local Differential Privacy Protection

1 Introduction

In recent years, social networks have obtained massive user information through websites and applications on users' smart devices, including e-commerce information data, medical diagnosis information data, national census data and financial business data [12, 19]. People rely on the network to access and collect all kinds of information, and to pass out their own information, but the process of information collection, storage, mining and exchange has the risk of privacy disclosure. In 2006, Dworku *et al.* [3] proposed the concept of differential privacy, which could realize privacy protection by adding interference to the user privacy data provided by the database system, and provide a privacy protection mechanism for users. At the same time, it also allowed statistical analysis for private data, such as mean value estimation and histogram estimation. Different from traditional privacy protection models such as k-anonymity [6] and l-diversity [11], differential privacy hardly assumes the background knowledge of the opponent, which provides a strong theoretical basis for the privacy of published data.

The ultimate goal of data management and data mining is data query and data release. Researchers have proposed many effective methods to protect the privacy of data release. Liang *et al.* [8] proposed the optimization-based k-anonymity method, which was mainly applied to the privacy protection of relational databases. This method generalized user records so that a single user record could not be distinguished from other $k-1$ records, thus preventing privacy leakage. The subsequent k-anonymity methods and their derivative technologies, such as l-diversity, t-closeness [7] and (a, k)-anonymity [18], are all proposed when the attacker has certain background knowledge. If the background knowledge is changed or is mastered by the attacker, these methods will no longer be applicable. Piao *et al.* [13] proposed a method for publishing equal-width histograms under differential privacy protection. Chen *et al.* [1] improved the reference [13] and increased the number of queries under the same privacy budget. However, histogram publishing method cannot solve query consistency problem. Therefore, some researchers improve the accuracy of the histogram of equal width. The post-processing method proposed by Zhu *et al.* [22] not only satisfied the query accuracy, but also reduced the addition of noise. However, the privacy budget cost of the above publishing strategy is high, and to solve this problem, the raw data is usually converted first, then the noise is added. These strategies mainly include histogram, wavelet change, Fourier transform, tree-based division and network-based division. The disadvantage of histogram differential privacy publishing method is that the number of queries allowed is limited, so it is necessary to design an efficient publishing algorithm to respond to query requests, so as to protect user data and prevent sensitive information from leaking.

With the development of various inheritance sensors and crowdsourcing systems, crowdsourcing information in various aspects can be collected and analyzed from various high-dimensional attributes to better generate rich knowledge related to groups, so that everyone in the crowdsourcing system can gain benefits. High dimensional heterogeneous data hides a lot of rules and information, mining these rules and information can provide better services for individuals and groups. High dimensional heterogeneous data is prone to dimensional disasters [4], and rich correlation is generated between data of different dimensions. This correlation facilitates data analysis and data mining. But as correlation increases, so does the dimension of the data, and the cost of processing these high-dimensional data increases exponentially. In differential privacy publishing of high-dimensional data, the relationship between high-dimensional data is complicated and simple linear processing cannot reflect the essential relationship between private data. Therefore, to release high-dimensional data under differential privacy, it is necessary to reduce the dimensions of privacy data and simplify the relationship between attributes, so that it is particularly important to control the sensitivity within a certain range.

Many scholars have studied the publishing of high-dimensional data sets that satisfy differential privacy. Zhang *et al.* [21] used classification tree to generalize high-dimensional data, determined the joint distribution of a group of approximate original data sets through attribute clusters, and determined the sample data. The PriView method [2] estimated the high-dimensional joint probability distribution by constructing the edge distributions of multiple low-dimensional attribute sets. This method provided effective privacy protection, but only applied to binary data sets. Xu *et al.* [17] presented DPPro method to publish high-dimensional data through random projection, which maximized practicality while ensuring privacy. The Lo Pub method [10] could identify the correlation between high-dimensional attributes, thus reducing the data dimension and improving the operational efficiency. Aiming at the problem of correlation between high-dimensional attribute sets, this paper proposes a local differential privacy protection theme of high dimensional data in embedded system based on hybrid differential swarm algorithm. In this paper, the maximum expectation algorithm is used to calculate the joint probability distribution of high dimensional data sets, and the Bayesian network is constructed based on normalized information entropy and mutual information, so that the constructed Bayesian network can restore the correlation of the original attribute set to a large extent. In this algorithm, the correlation between variables is obtained by mutual information method, and then the initial network is obtained by maximum spanning tree algorithm. If a node is arbitrarily designated as the root node of the undirected graph, a directed graph is automatically generated, and then the initial Bayesian network structure is obtained.

2 Related Works

2.1 Differential Privacy Protection Model

Differential privacy defines the limit of disclosure of user's personal data information in a specific database to the attacker or adversary, and achieves the effect of privacy protection by adding noise to the user's original data. Differential privacy ensures that even powerful attackers can limit their inferences about private information. Therefore, differential privacy is a feasible method to eliminate data privacy information leakage from data sources.

Definition 1. *(ε-difference privacy) For any two adjacent data sets D and D', given a privacy-protecting random algorithm M, $R(M)$ represents the set of all possible outputs of the random algorithm M, and S represents any subset of $R(M)$. If the random algorithm M satisfies:*

$$Pr[M(D) \in S] \leq Pr[M(D') \in S] \times e^{\varepsilon}. \tag{1}$$

Then it is said that random algorithm M satisfies ε-difference privacy [18].

In Formula (1), D represents a specific database. The ε is called the privacy budget and represents the privacy level provided by the random protection algorithm M on a particular database D, which determines the privacy protection intensity.

Definition 2. *(Sensitivity) For query $f : D \to R$, given adjacent data sets D_1 and D_2, sensitivity Δf of f is shown in Formula (2).*

$$\Delta f = max_{D_1,D_2}||f(D_1) - f(D_2)||_1. \tag{2}$$

Where $||\cdot||_1$ is L_1-normal form. Sensitivity is defined as the maximum impact that a single individual can have on the results of a data query. It represents the amount of noise that needs to be added to the result.

Local differential privacy is an improvement over traditional differential privacy protection. Traditional differential privacy defines the information boundaries of a user's data in a particular database that can be disclosed to third parties, and requires a trusted data collector. But local differential privacy does not require a trusted data collector [16]. Local differential privacy is used to solve the problem of private data for end users, and no data collector can access the complete real data set.

Definition 3. *(Local difference privacy) Given a random algorithm M, for any two user records X^i and $\hat{X}^i$, the probability of getting the same output X^* on the random algorithm M satisfies:*

$$P(M(X^i) = X^*) \leq e^{\varepsilon} P(M(\hat{X}^i) = X^*).$$

Then it is said that random algorithm M satisfies ε-local difference privacy [14].

2.2 Artificial Colony Algorithm

Artificial bee colony algorithm [20] is to simulate the collective cooperative behavior of bee colonies in the process of collecting honey. In this collaborative search model, a bee colony consists of three main parts: leading bees, following bees and scouting bees. During the search, points in the solution space are simulated as food sources when the swarm is looking for food. The process of collecting honey by artificial bees is also the process of searching for the best solution.

At the beginning, the bees are led to conduct a neighborhood search for the food source. If the fitness value of the searched solution is better than the previous one, the previous food source location will be replaced with the new one; otherwise, the original food source location will remain unchanged. After searching the location of the food source, all the leading bees return to the dance area and pass the nectar information on the food source to the following bees, who then select the food source to collect honey according to the generated probability based on the nectar information. For leading bee X_i and following bee V_i, according to the following formula,

$$V_i(j) = X_i(j) + r[X_i(j) - X_k(j)].$$

Update the food source location, where $i = 1, 2, \cdots, NP$. k is a random number not equal to i, $k = 1, 2, \cdots, NP$; $j = 1, 2, \cdots, D$. D is the dimension of solution space; $r \in (-1, 1)$ is a random number used to control the generation range of the $X_i(j)$ neighborhood.

The following bees select the food source by receiving the food source information conveyed by the leading bees, and then select the food source to collect honey according to its fitness value. Selection probability:

$$P_i = f_i / \sum_{i=1}^{NP} f_i.$$

Where, f_i is the fitness value of the $i-th$ solution. NP is the number of solutions, that is, the number of individuals in the colony who have completed the food source location update.

For each solution in the swarm algorithm, the update is counted by the corresponding variable with an initial value of zero. If the fitness value of a solution does not improve after successive *limit* times, it can be considered that the solution falls into local optimal. So the leading bee corresponding to this solution has to turn into a scouting bee. Assuming the solution is X_i, then the scouting bee passes:

$$V_i(j) = X_i(j) + rand(0,1)[X_{best}(j) - X_i(j)].$$

A new solution V_i is randomly generated to replace X_i and rejoin the colony for foraging, where X_{best} is the optimal individual in the colony.

3 Proposed Local Differential Privacy of High Dimensional Data

The problem of dimensionality reduction should be considered in publishing high dimensional data, which can be achieved by constructing Bayesian network. The low-dimensional edge distribution in Bayesian networks can accurately approximate the complete distribution of attributes in the corresponding high-dimensional data set. However, once the constructed Bayesian network is attacked by attackers, the privacy of users will be leaked. Therefore, it is necessary to add noise in the process of constructing Bayesian networks to generate disturbed data, so that the attacker will not cause the leakage of user privacy data when attacking Bayesian networks. Therefore, it is necessary to add noise to the constructed directed acyclic graph before constructing Bayesian networks.

3.1 Problem Description

Suppose there are n users, and each user has d attributes. The original user data set is $U = (u^1, u^2, \cdots, u^n)$, where u^i represents the data record of the $i-th$ user, and each user data has d attribute values, whose attribute values are expressed as $u^i = (u^i_1, u^i_2, \cdots, u^i_d)$. Attribute set $A = a^1, a^2, \cdots, a^d$, the value of each attribute is $a^i = (a^i_1, a^i_2, \cdots, a^i_d)$, where a^i_k represents the $k-th$ value of the $i-th$ attribute. Attribute value domain is $\Omega = \Omega_1, \Omega_2, \cdots, \Omega_d$.

3.2 Data Source Differential Privacy Protection

The purpose of local differential privacy protection for raw high-dimensional data sets is to prevent the data set from being attacked by attackers during transmission or on the server. Suppose that the original data of the $i-th$ user u^i is $u^i = (u_1^i, u_2^i, \cdots, u_d^i)$, the $i-th$ original attribute is $a^i = (a_1^i, a_2^i, \cdots, a_d^i)$. Setting the number of bits $(|\Omega_i|)$ in the string based on the range size of each attribute value. For each attribute a^i, the value of its corresponding bit is set to 1, the rest values of the bits are set to 0, and finally a binary string $\hat{s}_i$ of length $(|\Omega_i|)$ is formed.

After binary processing of the original data, the binary string s_j^i is obtained, and then each bit is answered randomly with the probability of f, and answered truthfully with the probability of $1-f$, as shown in Formula (3).

$$\hat{s}_j^i = \left\{ \begin{array}{l} 0 \ or \ 1 \text{ if } pro = f/2 \\ s_j^i[b] \text{ if } pro = 1-f \end{array} \right\} \tag{3}$$

Where, f is a user adjustable parameter to control the level of privacy protection.

The pseudo-code of the local differential privacy binary string conversion algorithm is as follows.

Algorithm 1 Local differential privacy binary string conversion algorithm

```
Input: user raw data set $u_j^i | j = 1, 2, \cdots, d$, original attribute set $A = a^1, a^2, \cdots, a^d$, random flip probability $f$.
Output: The randomized binary string $\hat{s}_i$ of the original data.
Begin
if $1 \le j \le d$ then
    Set the corresponding bit of the $j-th$ attribute value of the user $i$ to 1 and the other bits to 0 in the bit string, and finally convert it to the binary string $s_j^i$;
    Answer each bit of each bit string randomly with a probability of $f$, and answer truthfully with a probability of $1-f$;
    Finally, get a random flipped binary string $\hat{s}_j^i$;
end if
if Connect $d$ attributes to form a binary string, denoted as $\hat{s}_i$ then
    Return $\hat{s}_i$
end if
End
```

3.3 Hybrid Difference Bee Colony Algorithm for Bayesian Network Construction

A. Coding, Crossover and Mutation Operations

In this paper, the connection matrix between nodes $E = (C_{ij})$ is used to represent the network structure. $C_{ij} = 1$ means that node i points to node j. $C_{ij} = 0$ indicates that node i and node j are not connected. Each matrix is an individual for a Bayesian network, as shown in Figure 1. Its coding matrix is:

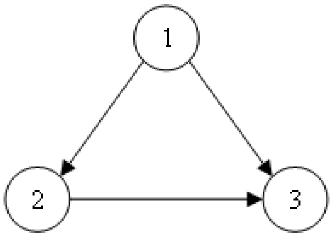


Figure 1: Structure of Bayesian network

$$E = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

The crossover process takes two particles to cross, and randomly selects their crossing positions. The elements at the intersection of the two individuals are exchanged at the given intersection rate. In the mutation process, a position is randomly selected according to the adaptive adjusted mutation rate, so that C_{ij} changes from 1 to 0 or from 0 to 1 (that is, edge addition, edge reduction and reverse operation of the network structure).

B. The Generation of Initial Population Structure

Average mutual information is the difference between a priori uncertainty and a posteriori uncertainty under the meaning of statistical average, which is the statistical average of mutual information. Calculate the mutual information between the two nodes to know whether the two variables are related. There must be no causal relationship between those variables that do not exist correlation, which can initially narrow the search space. The mutual information between nodes is calculated as follows:

$$I(X,Y) = \sum_{i,j} P(x_i, y_j) log_2 [\frac{P(x_i, y_j)}{P(x_i), P(y_j)}].$$

Where, $I(X,Y)$ represents the mutual information between variables X and Y. x_i and y_j represent the values of the variables X and Y respectively. $P(x_i, y_j)$ denotes the probability between x_i and y_j.

According to the mutual information, the weight of each node is calculated, and then a candidate network is generated by using the method of maximum spanning tree. In this case, the alternate network is an undirected graph, and then any node is designated as the root node and a tree with this node as the root

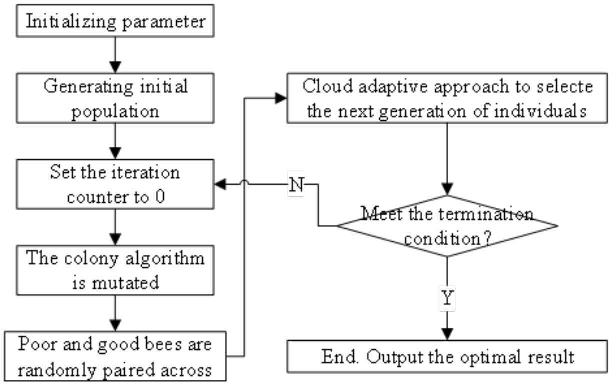

Initializing parameter
Generating initial population
Set the iteration counter to 0
The colony algorithm is mutated
Poor and good bees are randomly paired across
Cloud adaptive approach to selecte the next generation of individuals
Meet the termination condition?
N
Y
End. Output the optimal result

Figure 2: Hybrid differential swarm algorithm

node is generated. Finally, we randomly add edges, subtract edges and reverse operations between nodes other than the tree of the alternate network, and select the reasonable network structure as the initial population.

C. Scoring Function

The scoring function is to measure the matching degree between the generated Bayesian network structure and the data sample. In this paper, the Bayesian Information Criterion (BIC) [5], a scoring function commonly used in Bayesian network structure learning algorithms, is selected as the fitness function. The larger fitness value denotes the higher matching degree. The expression formula is:

$$Q_{BIC} = LL(B|D) - \frac{1}{2log_2N}Dim(B).$$

Here,

$$LL(B|D) = \sum_{i=1}^{n} log_2P(B|D).$$

It is a measurement of the number of bits needed to describe D. $1/(2log_2N)$ indicates the number of bits used for each parameter.

$$Dim(B) = \sum_{i=1}^{n}(r_i - 1)q_i. \quad (4)$$

Formula (4) is the dimension of the Bayesian network, i.e., the number of required free parameters.

D. Description of Hybrid Differential Swarm Algorithm

Figure 2 shows the process of hybrid differential swarm algorithm.

1) Mutation behavior. The colony algorithm is applied to the mutation behavior of differential evolution algorithm to improve it. The main search is based on the behavioral changes of the leading bees, the following bees and the scouting bees. For each individual X_i and the fitness f_i of each individual in the population X to be optimized, the mutation rate is generated by adaptive method to accelerate the convergence of the population.

   For leading bee X_i and following bee V_i, the food source location is updated according to the following rules. Let $\bar{f}$ be the average fitness value of the current population, then individuals whose fitness value is lower than $\bar{f}$ will be mutated with the variation rate P_{m1}. Individuals with fitness value higher than $\bar{f}$ are mutated with variation rate P_{m2}, and fitness values are calculated respectively. The variation rate P_{m1} is generally set at [0.2,0.3], because individual fitness values at this time are lower than the average value, so a larger variation rate is used. Similarly, variation rate P_{m2} is generally set at [0.1-0.2], because individual fitness values at this time are higher than the average value.

   The choice of food source for the following bees is based on the rate of return, i.e. $P_i \geq rand(0,1)$, then the following bees after the mutation of the leading bees will be identified as effective mutation; otherwise, the following bees will be identified as invalid. The individuals before the mutation will continue to search for food, and the fitness values of each individual will be calculated after the selection is completed.

2) Crossover behavior. For each individual X in the modified population X_i, it finds individual X' whose fitness value is lower than the average fitness value $\bar{f}$, and individual X'' whose fitness value is higher than the average fitness value $\bar{f}$. Some individuals are randomly selected from X' and X” to cross probabilities P_c. The crossover rate P_c is generally set as [0.1,0.3]. P_c is set to a large range because there is too much difference between individuals, so there is a big room for choice according to the actual situation.

3) Selection behavior. For each individual X_i and the fitness f of each individual in the population X to be optimized, the following cloud adaptive method is adopted to generate crossover rate to accelerate the convergence of the population.

   Let $\bar{f}$ be the average fitness value of the current population. The fitness value of the optimal particle is f_{max}. f_{avg1} is obtained by averaging fitness values below $\bar{f}$. f_{avg2} is obtained by averaging the fitness value higher than $\bar{f}$. When the population individual has a small fitness value, a small acceptance probability CR is adopted to search for a structure better than the current structure as far as possible to accelerate the convergence of the population. When the population individual has a high fitness value, a larger acceptance probability CR is taken to retain the better individuals and a certain amount of inferior population as far as possible, so as to increase the diversity of the population. Its generation rule is that when f_i is lower than f_{avg1}, $CR = 0.4$; When f_i is better than f_{avg2}, $CR = 0.9$; Otherwise, the value is as follows:

$$\begin{aligned} E_x &= f_{avg2}, E_n = \frac{|f_{max} - f_{avg2}|}{c_1}. \\ H_e &= E_n/c_2. \\ E'_n &= random(E_n, H_e). \\ CR &= 0.3 + 0.5exp(-\frac{(f_i - E_x)^2}{2E'^2_n}). \end{aligned}$$

Where $c_1 = 2.8$ and $c_2 = 10$ are control coefficients. When $CR \geq rand(0,1)$, the individual is selected for the next generation evolution; otherwise, the individual that fails to pass the selection is replaced with the structure after crossing and enters the next generation evolution as a whole. In order to accelerate the convergence rate, the crossover rate P_b at this time is generally selected from [0.05,0.15]. The characteristics of the optimal individual can be obtained by using a small crossover rate, and the most important thing is not to destroy the original individual, so as to maintain the differences between individuals.

4 Experimental Results and Analysis

In this section, the proposed algorithm in this paper will be verified and explained through specific experiments. The experiment is based on python platform and Pytorch framework, GPU RTX3060Ti. MNIST and CIFAR-10 standard data sets are used in the experiment. Data availability of differential privacy in Bayesian networks is generally determined by comparing training accuracy under the same privacy budget. Set $\delta = 10^{-5}$, and the privacy budget is $\varepsilon = [1, 2, 3, 4, 5, 6, 7, 8]$.

In this section, we conduct comparison experiments with other methods including Fedsel [9], LDP-Fed [14], DP-cryptography [15]. Figure 3 and Figure 4 show that the training accuracy of algorithm LDP-Fed is low, which may be because the model noise is added in a single way, and the global sensitivity is considered to calculate the Gaussian noise required by the gradient. Firstly, LDP-Fed considers the internal changes in the process of gradient descent, and realizes differential privacy mechanism by projecting gradient into auxiliary low-dimensional subspace. Under high privacy requirements, LDP-Fed performs better than Fedsel. However, with the decrease of noise, it may introduce more reconstruction errors and lead to the decline of accuracy. DP-cryptography pre-sets the noise group, adopts double noise interference, and has A more detailed noise control. However, only using regularization to improve the training accuracy is better than that of Fedsel and LDP-Fed. The proposed algorithm has different optimizations in sensitivity, privacy budget allocation and gradient clipping, so the training accuracy is improved. When $\varepsilon = 2$, the increased values in MNIST and CIFAR-10 data sets are 1.83% and 8.52%.

It is not difficult to find from Table 1 that under the same privacy budget, the selection of k value is not very sensitive to the influence of training accuracy. As long as the selection of k value can make the decomposition matrix contain enough features of the original gradient matrix without introducing too much gradient redundancy, the training will tend to be stable. Therefore, in the following experiment, this paper selects $k = 50$ for experiment, and compares it with other adaptive algorithms under the same network environment and privacy budget.

5 Conclusions

In this paper, the differential evolution algorithm is mixed with bee colony algorithm, and the improved methods such as cloud adaptive theory are introduced into each stage of the hybrid algorithm to form a cloud adaptive hybrid algorithm. At the same time, this algorithm is applied to the Bayesian network structure learning method. Through the simulation experiment of the classical network model, it is proved that the algorithm has better optimization ability. This algorithm can effectively improve the convergence speed and avoid falling into local optimality. Through experiments, it can be seen that the learning effect of the network is significantly enhanced when the number of iterations or data set increases. This algorithm has some reference value to the application of Bayesian networks in practical engineering.

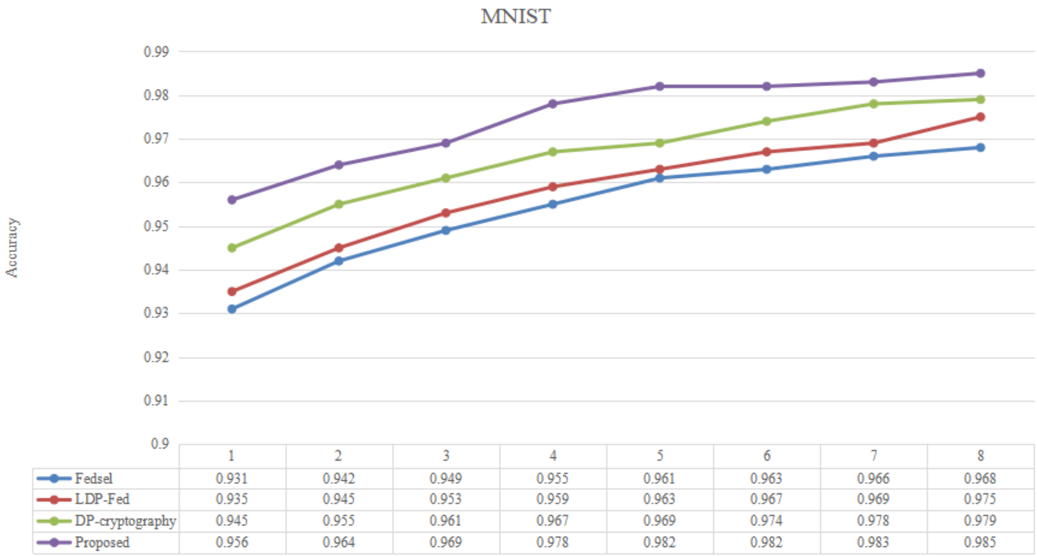
Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

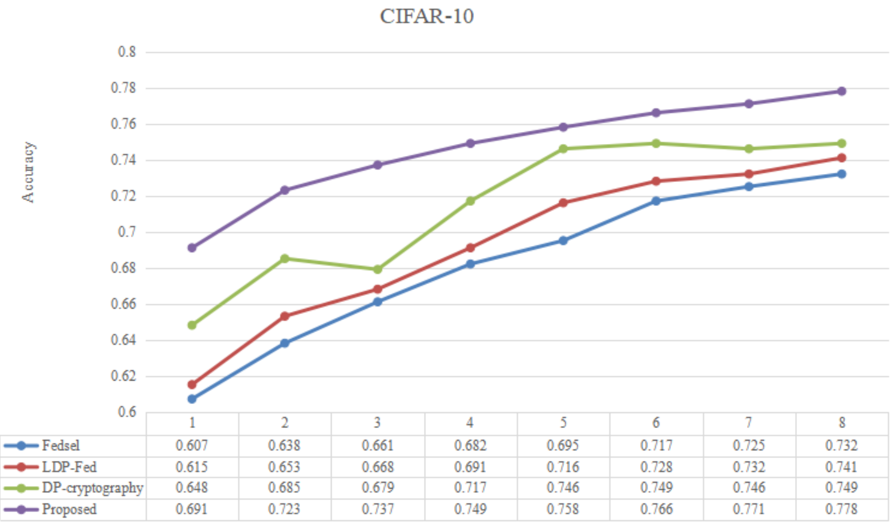
[1] Q. Chen, Z. Ni, X. Zhu, *et al.*, "Differential privacy histogram publishing method based on dynamic sliding window," *Frontiers of Computer Science*, vol. 17, no. 4, pp. 174809, 2023.

[2] X. Cheng, P. Tang, S. Su, *et al.*, "Multi-party high-dimensional data publishing under differential privacy," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 8, pp. 1557-1571, 2019.



	1	2	3	4	5	6	7	8
Fedsel	0.931	0.942	0.949	0.955	0.961	0.963	0.966	0.968
LDP-Fed	0.935	0.945	0.953	0.959	0.963	0.967	0.969	0.975
DP-cryptography	0.945	0.955	0.961	0.967	0.969	0.974	0.978	0.979
Proposed	0.956	0.964	0.969	0.978	0.982	0.982	0.983	0.985

Figure 3: Comparison of training accuracy under different differential privacy conditions in MNIST



	1	2	3	4	5	6	7	8
Fedsel	0.607	0.638	0.661	0.682	0.695	0.717	0.725	0.732
LDP-Fed	0.615	0.653	0.668	0.691	0.716	0.728	0.732	0.741
DP-cryptography	0.648	0.685	0.679	0.717	0.746	0.749	0.746	0.749
Proposed	0.691	0.723	0.737	0.749	0.758	0.766	0.771	0.778

Figure 4: Comparison of training accuracy under different differential privacy conditions in CIFAR-10

Table 1: The training accuracy of different k

Data set	Privacy budget	10	50	200	500
MNIST	$(2,10^{-5})$	96.23	96.48	96.52	96.18
MNIST	$(4,10^{-5})$	97.44	98.97	98.83	98.46
MNIST	$(8,10^{-5})$	98.27	98.48	98.46	97.99
CIFAR-10	$(2,10^{-5})$	72.70	73.26	72.27	71.38
CIFAR-10	$(4,10^{-5})$	74.96	75.94	75.63	75.16
CIFAR-10	$(8,10^{-5})$	75.88	77.89	76.98	76.25

[3] C. Dwork, "Differential privacy," in *33rd International Colloquium on Automata, Languages and Programming (ICALP'06)*, pp. 1-12, 2006.

[4] E. Furlan, P. Dalla Pozza, M. Michetti, *et al.*, "Development of a Multi-Dimensional Coastal Vulnerability Index: Assessing vulnerability to inundation scenarios in the Italian coast," *Science of The Total Environment*, vol. 772, pp. 144650, 2021.

[5] J. Hu, H. Qin, T. Yan, *et al.*, "Corrected Bayesian information criterion for stochastic block models," *Journal of the American Statistical Association*, vol. 115, no. 532, pp. 1771-1783, 2020.

[6] H. N. S. S. Jagarlapudi, S. Lim, J. Chae, *et al.*, "Drone helps privacy: Sky caching assisted k-anonymity in spatial querying," *IEEE Systems Journal*, vol. 16, no. 4, pp. 6360-6370, 2022.

[7] N. Li, T. Li, S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *IEEE 23rd international conference on data engineering*, pp. 106-115, 2006.

[8] Y. Liang, R. Samavi, "Optimization-based k-anonymity algorithms," *Computers & Security*, vol. 93, pp. 101753, 2020.

[9] R. Liu, Y. Cao, M. Yoshikawa, *et al.*, "Fedsel: Federated sgd under local differential privacy with top-k dimension selection," in *25th International Conference on Database Systems for Advanced Applications*, pp. 485-501, 2020.

[10] C. Lo, J. P. Lynch, M. Liu, "Distributed reference-free fault detection method for autonomous wireless sensor networks," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 2009-2019, 2013.

[11] Z. Lv, F. Piccialli, "The security of medical data on internet based on differential privacy technology," *ACM Transactions on Internet Technology*, vol. 21, no. 3, pp. 1-18, 2021.

[12] A. S. T. Olanrewaju, M. A. Hossain, N. Whiteside, *et al.*, "Social media and entrepreneurship research: A literature review," *International Journal of Information Management*, vol. 50, pp. 90-110, 2020.

[13] C. Piao, Y. Shi, J. Yan, *et al.*, "Privacy-preserving governmental data publishing: A fog-computing-based differential privacy approach," *Future Generation Computer Systems*, vol. 90, no. 158-174, 2019.

[14] S. Truex, L. Liu, K. H. Chow, *et al.*, "LDP-Fed: Federated learning with local differential privacy," *Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking*, pp. 61-66, 2020.

[15] S. Wagh, X. He, A. Machanavajjhala, *et al.*, "DP-cryptography: Marrying differential privacy and cryptography in emerging applications," *Communications of the ACM*, vol. 64, no. 2, pp. 84-93, 2021.

[16] Y. Wang, Y. Tong, D. Shi, "Federated latent dirichlet allocation: A local differential privacy based framework," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 4, pp. 6283-6290, 2020.

[17] C. Xu, J. Ren, Y. Zhang, *et al.*, "DPPro: Differentially private high-dimensional data release via random projection," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 12, pp. 3081-3093, 2017.

[18] M. Yang, I. Tjuawinata and K. Y. Lam, "K-Means Clustering With Local d_x-Privacy for Privacy-Preserving Data Analysis," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2524-2537, 2022.

[19] M. Yang, I. Tjuawinata, K. Y. Lam, J. Zhao and L. Sun, "Secure Hot Path Crowdsourcing With Local Differential Privacy Under Fog Computing Architecture," *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 2188-2201, 2022.

[20] S. Yin, J. Liu, L. Teng, "An Improved Artificial Bee Colony Algorithm for Staged Search," *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, vol. 14, no. 3, pp. 1099-1104, 2016.

[21] W. Zhang, J. Zhao, F. Wei, *et al.*, "Differentially private high-dimensional data publication via Markov network," *EAI Endorsed Transactions on Security and Safety*, vol. 6, no. 19, 2019.

[22] K. Zhu, P. Van Hentenryck, F. Fioretto, "Bias and variance of post-processing in differential privacy," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, pp. 11177-11184, 2021.

Biography

Shiru Sun biography. Sun Shiru (1987.08), male, Han nationality, born in Yantai, Shandong Province, teacher of Scientific Research Department of Zhengzhou University of Science and Technology, master candidate, lecturer. Research direction: Embedded system, Industrial robot.

Bin Wang biography. Bin Wang is with College of Big Data and Artificial Intelligence, Zhengzhou University of Science and Technology. His majors are big data and AI.

Hongwei Zhang biography. Hongwei Zhang is with College of Civil Engineering and Architecture, Zhengzhou University of Science and Technology. The interests include Information fusion, data mining.

Data Security Analysis Based on ReXNet Network With Rule-based Reasoning for Intelligent Cooperative Robot Design

Yu Jiang
(Corresponding author: Yu Jiang)

Shenyang Normal University
Shenyang 110034, China
Email: snowycry@qq.com

(Received Dec. 20, 2022; Revised and Accepted Apr. 16, 2023; First Online Apr. 30, 2023)

The Special Issue on Computational Intelligence Networks for Privacy and Security in Evolving Internet of Multimedia Things
Special Editor: Prof. Shoulin Yin (Harbin Institute of Technology)

Abstract

The evaluation information in an intelligent, cooperative robot system has a single source and a significant accuracy deviation problem. At the same time, the fusion of robot data and heterogeneous data is insufficient. Therefore, we propose a novel data security analysis structure combining the ReXNet network with rule-based reasoning. Meanwhile, we apply this structure to the design of an intelligent, cooperative robot. We build a ReXNet network model based on attack behavior to learn features and reconstruct the security data of intelligent, cooperative robots. A data fusion based on rule reasoning is proposed to improve robot safety data's classification ability. Experimental analysis shows that the ReXNet network with a rule-based reasoning model further improves the efficiency of data security in intelligent, cooperative robots.

Keywords: Data Security Analysis; Heterogeneous Data; Intelligent Cooperative Robot System; ReXNet Network; Rule-based Reasoning

1 Introduction

In the context of big data, the construction and application of intelligent cooperative robot platform continues to develop deeply [8,9]. While effectively serving all kinds of infrastructure, intelligent applications and data resources, it also puts forward new requirements for the overall security of the platform. With the continuous expansion of the scope and scale of intelligent cooperative robot network [3], the platform security is always threatened by virus invasion, denial of service attack, 5G cluster attack, injection attack, malware and other problems [14]. Intelligent collaborative robots provide cloud-based security through security devices across endpoints, the cloud, and the Web. Using security logs generated by these devices directly for network intrusion detection faces three kinds of problems [16]. a) There are a large number of heterogeneous devices with different data formats, diverse standards, and complex protocols. It is difficult to achieve normalization of these data, and security logs of multiple devices cannot be fused and detection results of a single device are inaccurate; b) Intrusion detection has no content-oriented and feature-oriented secure heterogeneous big data clustering and fusion method, which relies heavily on expert knowledge and cannot be qualitatively evaluated; c) Security information in intelligent collaborative robot data environment is difficult to set up a common semantic reference model in advance due to complex data structure and dynamic node increase or decrease [5]. How to detect attack events quickly and accurately, achieve double pressure drop of security event storage and total calculation, and improve the quality and timeliness of security analysis data have become the main challenges facing intelligent robot network security today.

At present, ReXNet, as a deep learning method, has made great contributions to image recognition, speech recognition and machine vision [7]. Intrusion detection combining with ReXNet has been widely used in network threat detection. When analyzing the security data of intelligent cooperative robots, it is found that each threat behavior data set has one-dimensional feature similarity. Therefore, this paper establishes a data classification model based on ReXNet for security data processing. Firstly, based on the security data set, the attack technology mechanism and the characteristics of the attack target are summarized systematically, and an extensible attack behavior model is established. Then, based on the attack behavior model, the ReXNet model is built to learn and reconstruct the features of the security data, and the hierarchical mining of the time hidden features in the data set solves the shortcoming of insufficient feature extraction of the existing decision-level data. Finally, the multi-source heterogeneous security data with attack mode as the core is normalized into threat events, and the classification effect of intrusion detection is further enhanced. The flexible scheme is constructed for the fine-grained security state detection, and further provides a reference for the data security protection ability of robots. The main contributions of this model are as follows: a) it uses multi-source logs from intelligent cooperative robots, including IDS logs, switching device logs and system logs as intrusion detection data sets, achieves better results than single source logs as intrusion detection data sets; b) we propose a ReXNet network model based on the robot data environment, extract influential features from the intrusion data of a single evidence source for feature extraction and reconstruction, and obtain a well-classified model through training, achieving a high detection rate and recall rate; c) An attack fusion model is established. Based on the ReXNet model, rule reasoning fusion processing is carried out on heterogeneous data to further improve the accuracy and greatly improve the current intrusion detection and identification efficiency of robot data security.

2 Related Works

2.1 ReXNet Model

ReXNet is improved based on the lightweight MobileNetV2 network with appropriate adjustments to effectively reduce the existing network feature performance bottleneck. There are Inverted Residuals, Linear Bottleneck and SE in SENet (Squeeze-and-Excitation Networks) are the important bases for the lightweight neural network MobileNetV2 [12]. ReXNet combines the improvement of increasing the number of network channels, replacing the activation function to Swish-1 function, and designing more extension layers to reduce the bottleneck problem of the network feature performance, thus forming the basic model structure of ReXNet lightweight network.

The ReXNet model is the same as most CNN models in terms of data processing. After optimizing the algorithm, it improves the data processing capacity, speeds up the computational efficiency and effectively saves computing resources. The idea of improving the rank of data in network module solves the problem of extracting image features completely without compression as far as possible, which runs through the design of lightweight neural network [15]. A key building block of ReXNet lightweight network is the reciprocal residual structure composed of deep separable convolution. Its basic idea is to replace complete convolution operators by decomposed convolution operators and to use a small number of operators and operations to achieve the same computational effect.

The reciprocal residual structure can effectively avoid the problem of feature information loss when the parameters of conventional convolution kernel appear more than 0, that is, the convolution kernel does not play the role of feature extraction network. More feature data information can be obtained by using the reciprocal residual structure, so as to improve the training effect of the model. In the network structure, the inverted residual structure mainly adopts the operation of first raising dimension, that is, first expanding the extension layer, and the expansion multiple is controlled by the extension factor. At this time, the activation function of the dimension raising convolutional layer is ReLU6, with the main purpose of obtaining more feature extraction information. Then feature extraction is carried out on Depthwise Convolution (DW) [13]. At this time, activation function of the convolutional layer of feature extraction is ReLU6. Finally, the convolution process of dimension reduction compression is carried out, and the activation function is linear activation function. The overall network structure is small at both ends and large in the middle. This is also very different from the residual structure, the two present completely opposite structure, so it is called the reciprocal residual structure.

Separable convolution is divided into space separable convolution and depth separable convolution. The core convolution layer of ReXNet network is depth-separable convolution. Deep separable Convolution splits ordinary convolution into deep convolution and Pointwise Convolution. Deep convolution performs lightweight filtering by applying a separate convolution filter to each input channel. Point-by-point convolution constructs new features by linear combination of input channels to achieve the raising and lowering of dimension of feature graphs.

2.2 Semantic Query Based on Rule Reasoning

Semantic retrieval is the retrieval of one or more ontologies [4]. First, the retrieved ontology should be identified as shown in Figure 1. Therefore, all domain ontology indexes should be stored, and all relevant ontologies should be found according to the keywords searched, so as to prepare for the next semantic retrieval. If there is only one ontology, then the semantic retrieval is carried out based on the ontology. If several ontologies are found, the fields involved are selected in the form of question and answer. When matching, not only the type of query is taken as the main matching item, but also the parameters need to be matched, and the ontology similarity is calculated to list the relevant ontology. The specific similarity $sim(i,o)$ is:

$$sim(i,o) = ui_k + \frac{\sum_{j=1}^{n} ri_j}{n} \tag{1}$$

The left side of the equal sign represents the input i and ontology o approximation of the keyword. i_k indicates that keywords that can express the type can be retrieved in ontology o. i_j represents whether the parameter can be found in the keyword semantics. If it can be found, it is $i_j = 1$; otherwise, it is 0. n is the number of parameters. u represents the coefficient of the keyword. sim stands

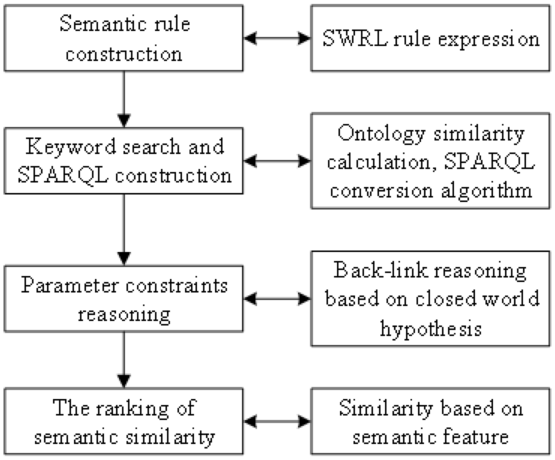

Semantic rule construction
SWRL rule expression
Keyword search and SPARQL construction
Ontology similarity calculation, SPARQL conversion algorithm
Parameter constraints reasoning
Back-link reasoning based on closed world hypothesis
The ranking of semantic similarity
Similarity based on semantic feature

Figure 1: Semantic retrieval procedure

for keyword and compactness matched by weight. r represents the correlation coefficient between the parameter and the target ontology under i_k, $u + r = 1$.

The formula of semantic sorting is:

$$Sim(A, B, O) = \frac{C_{super}(A, O) \cap C_{super}(B, O)}{minP_{super}(A, B, O)} \quad (2)$$

In the formula, $C_{super}(A, O)$ and $C_{super}(B, O)$ represent the number of concepts A and B in the ontology tree O. $C_{super}(A, O) \cap C_{super}(B, O)$ represents the number of intersections. $minP_{super}(A, B, O)$ represents the shortest path of concepts A and B in the ontology diagram. u, v and ϕ represent the coefficients of three terms. The coefficient u represents the coefficient of concept similarity degree of concepts A and B to the parent of ontology tree O. The coefficient v represents the coefficient of concept similarity degree of concepts A and B to child of the ontology tree. t represents the constraint satisfaction coefficient of the parameter. If there are many parameters, the ϕ value is relatively large, and the sum of the three parameters is 1, that is, $u + v + \phi = 1$.

in formula 2, $0 < Sim(A, B, O) < 1$, when $A \equiv B$, $Sim(A, B, O) = 1$. If it is completely unrelated or has no common parent concept, child concept and common parameter, $Sim(A, B, O) = 0$.

3 Proposed Fusion Model

In this paper, the fusion framework of ReXNet and rule-based reasoning based on attack pattern is proposed to transform network attacks into structured flags and use first-order logic relation to describe attacks. According to the proposed fusion model, the intention of the attacker can be extracted, and the security data fusion detection of intelligent robots can be effectively carried out in the case of large-scale data.

3.1 Definition of Attack Module

For intelligent collaborative robot data, it is necessary to collect all data related to security threats as far as possible, including risk data, attack event data, security state data, behavior analysis data and other internal security data, as well as a variety of external offensive and defense dynamics, attack samples, hacker attacks and other intelligence data. And through the perfect association analysis and data mining methods to discover the hidden value of these data, it can ensure the openness of the platform, compatible and support more security data. For example, a robot system will generate a large number of logs during operation. Since the system is in normal working state most of the time, the collected logs will have a large amount of redundant data [17]. If such security data is directly used, the interference degree of log analysis will be increased. Therefore, the redundant data in security data can be deleted by the establishment of attack mode. The effectiveness of attack behavior modeling depends on the precise modeling of a single attack action. By studying multiple attack behavior modeling methods and attack classification standards, a complete description of the attack behavior should include the following three dimensions: a) characteristics of the attack target, target type, target product technology, target relying operating environment, that is, attacks launched on specific platforms have strong target correlation characteristics; b) Attack mechanism, including initial conditions, resource requirements, attack means, attack level, time characteristics, severity, etc.; c) Attack intention, attack phase intention and expected attack result. For example, sniffer attacks can be described in three dimensions: the attack targets are network devices, and the running environment is broadcast networks; The attack mechanism is data flow analysis of network link flow, the initial condition is low security network structure, the resource demand is the same network segment, the attack means is penetration traffic analysis, the attack level is reconnaissance, the time characteristic is long-term high intensity, and the severity is to destroy the confidentiality of information. In the attack phase, the intention is to implement data flow analysis attacks, and the expected attack result is to implement replay attacks.

Based on the above three-dimensional method, this paper defines the attack mode and constructs the structural attack mode as shown in Figure 2. In order to establish the mapping relationship between the normalized attack situation data field and the unknown attack situation data field, three main information types of target characteristics, attack mechanism and attack intention in the attack data in the network attack situation should be extracted. The fusion of data structure and data content is the key to secure and heterogeneous big data semantic fusion. Traditional heterogeneous data semantic fusion methods usually need to set a common semantic reference model in advance. Security information in robot data environment is difficult to set up in advance because of complex data

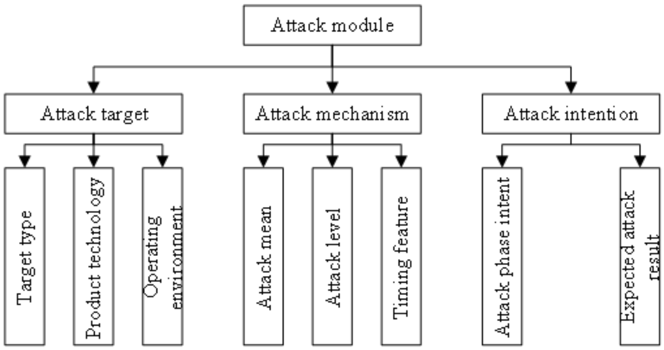


Figure 2: Definition of attack mode

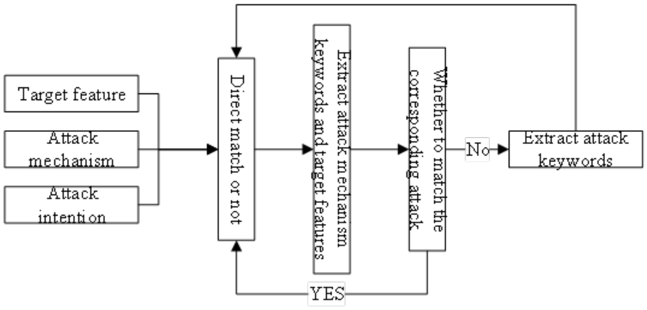

Target feature
Attack mechanism
Attack intention
Direct match or not
Extract attack mechanism keywords and target features
Whether to match the corresponding attack
No
Extract attack keywords
YES

Figure 3: Attack mode matching flow

structure and dynamic node increase and decrease. In this paper, dynamic semantic fusion mechanism is considered to conduct attack matching oriented to content features, so as to improve the confidence based on security events. The attack matching process is shown in Figure 3. If the extracted attack features and target features cannot match the attack mode, the attack intent needs to be further extracted to complete the classification of the attack mode.

3.2 Data Fusion Framework

There are a large number of heterogeneous devices in intelligent cooperative robots. These data have different formats, diverse standards and complex protocols. The difficulty of multi-source heterogeneous security data fusion lies in that log data collected by different security suppliers and devices in intelligent cooperative robots are inconsistent in describing network attacks and describing the same situation information, so it is difficult to achieve the normalization of these data [10]. This section proposes a data fusion framework based on intelligent cooperative robots, as shown in Figure 4. First, it collects and parses data from heterogeneous network devices, security devices, system logs and other data sources. Then it filters and aggregates the repeated logs collected by the holding pair through the self-defined attack mode to reduce the amount of logs. Then the data are preprocessed, such as random sampling, mean normalization, one-hot coding, etc. Then the feature data is combined to divide the training set and the test set, and sent into the ReXNet model for data classification. Data sources of different heterogeneous situations need to be integrated to extract and describe abstract attack intentions. Therefore, data classification is carried out concurrently. Data of different devices are classified by ReXNet and sent into the rule reasoning module for security data fusion. Finally, the attack type is identified.

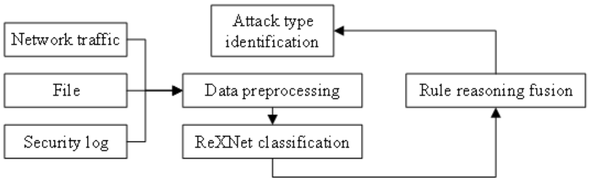


Figure 4: Data fusion framework

The ReXNet model based on intelligent cooperative robot is shown in Figure 5. Where the input data sets are N sensors X. X are N independent K-dimensional sample sets. The model used in this paper consists of two convolution layers, one pooling layer, two global pooling layers and one fully connection layer. The specific steps are as follows:

- Data preprocessing. When the data is input, the data is preprocessed and one-hot coding is adopted to map the data to [0,1], so as to reduce the impact of the inconsistency of eigenvalues on the results.

- Convolving-pooling-pooling operation. After convolution layer $convld_1$, it executes pooling operation to reduce the number of features. The output is $T^{lth}_{mth} = ReLU(maxpooling(S^{lth}_{mth}, P^{lth}_{mth}) + b^{lth}_{mth})$. Where, b^{lth}_{mth} is the deviation layer of lth convolution, and the step length is 1. The output is then fed into the next convolution layer $convld_2$. And it saves the feature graph Y' of $convld_2$. Where the operation for each element Y' in y' is defined as $y' = ReLU(\sum_{i\in[0,l]} input(i+c, channel) \times C_j + b)$. l is the length of feature graph. i is the sequence number of the convolution kernel. c is the length of the convolution kernel. C_j represents the length of the convolution kernel whose sequence number is j.

- It conducts global average pooling and global max-pooling respectively on the obtained results. In this part, $1 \times k$ feature maps are obtained. In order to prevent over-fitting, softmax is used to obtain classification results at the fully connection layer. $y = soft(y')$ is calculated for each element y, and loss function is$loss = crossentropy(Y, Y')$.

- Adam optimizer is applied to optimize losses. When the termination condition of training iteration is reached, weight update and bias update are carried out, and batch normalization is used to normalize

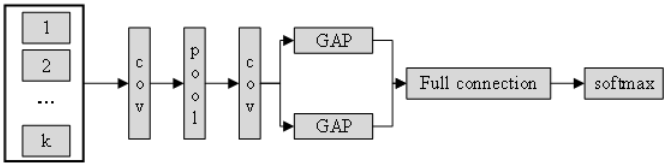


Figure 5: ReXNet model in this paper

each feature during training. Then the entire training data set is re-scaled again to make the training convergence faster and improve high performance results. In order to prevent over-fitting phenomenon, the dropout function is used to delete the output scale of the upper layer to provide regularization when merging the global pooling results, and the invisible results are generalized. Some neurons are randomly ignored to prevent over-fitting.

In order to solve the problem of rationality and validity of the fusion results of multiple safety equipment bodies in intelligent cooperative robots under the condition of uncertain synthesis rules, it combines the rule reasoning and the safety data classification model. Identification framework θ contains N-dimensional row vector $S_i(i \leq k)$ with n different conclusions and M different information sources. The probability assignment function of the vector distribution is set as $k_1, k_2, \cdots, k_K$. m_i is the distribution probability of the corresponding element. A_i is the ith evidence in order to be able to correlate the correlation between evidence and distance. For the evidence source S_i, S_j, $D(S_i, S_j)$ is a $2N \times 2N$ matrix between them. The distance between two sources of evidence is defined as $d(S_i, S_j)$.

$$D(S_i, S_j) = \frac{S_i \cap S_j}{S_i \cup S_j} \tag{3}$$

$$d(S_i, S_j) = \sqrt{\frac{(S_i - S_j)^T D(S_i - S_j)}{2}}. \tag{4}$$

Formula (4) is the degree of correlation between evidence S_i and S_j. The value of $d(S_i, S_j)$ is [0,1], that is, the correlation degree can be set as $cor(S_i, S_j) = e^{1-d(S_i,S_j)}$. The closer $cor(S_i, S_j)$ is to e, the strong correlation between the evidence is shown. Meanwhile, the total support degree of S_i is the sum of the support degree supported by other evidence, which can be expressed by the $cor(S_i, S_j)$ function as $sup(S_i) = \sum_{j=1}^{M} cor(S_i, S_j), i \notin M$. If the credibility weight of evidence is highly similar to the average support degree, it means that the evidence is supported by most other evidence and has high credibility. The credibility weight for each piece of evidence is $t = sup(S_i)/\sum_{j=1}^{M} sup(S_j)$. On this basis, the credibility of evidence is determined according to the evidence source. Total credibility is $T = (\frac{\sum_{i=1}^{M-1}\sum_{j=1}^{M} cor^{M!}(S_i,S_j)}{M!})^{\frac{1}{M!}}$.

4 Experiment and Analysis

The data set used in the experiment in this paper is the intrusion data set collected by the server placed in the intelligent robot and its corresponding firewall, file system, network equipment, etc. The data set mixes normal working activity with synthesized simulated attack behavior. According to the above definition of the attack mode, the data set can be divided into 10 attack types [1], namely Normal, Fuzzers, Reconnaissance, Backdoors, DoS, Exploit, Analysis, Generic, Shellcode and Worms. The specific data set is described in Table 1. A total of 41 features are generated, which are packet attribute-based and stream attribute-based. Packet attribute-based helps check the load in the packet header. Stream-based features are direction, arrival time, and length, such as srip, sport, dstip, dsport, and proto. The random sampling technique is used to solve the problem of data imbalance between the number of attacks and the normal data.

In this paper, PyTorch is used to build the ReXNet model. Firstly, data is preprocessed according to the definition of attack mode, and then feature extraction is carried out, including random sampling, mean normalization, one-hot coding and other methods. Different dimension reduction methods are used to remove redundant and irrelevant features from network traffic data, and then the characteristic data is combined to divide the training set and test set. The results are obtained by input into the model respectively. The specific parameters are shown in Table 2.

Precision, recall and F1 score are used as evaluation criteria. *Precision* is the precision rate, that is, the proportion of the number of correctly classified attack samples in the total number of samples; *Recall* is the percentage of the predicted correct samples in the actual normal samples. $F1$ score is the harmonic average of the accuracy rate and recall rate of the model. FP (false positive) is the wrong sample judged as the correct sample; FN (false negative) is the correct sample judged as the wrong sample,TP (true positive) is the correct sample judged as the correct sample. The calculation formula is as follows:

$$P(Precision) = \frac{TP}{TP + FP}.$$

$$R(recall) = \frac{TP}{TP + FN}.$$

$$F1 = \frac{2PR}{P + R}.$$

The proposed model in this paper is compared with other methods including MLDL [2], IMIDS [6], CGOA [11] in the same data set, and the results are shown in Table 3. The P and R of the model are more than 98%. After dimensionality reduction, the low-dimensional characteristic data set realizes the redundancy removal in network traffic, and it can be concluded that the proposed model has better advantages in feature selection and detection results.

Table 1: Description of intrusion dataset

Type	Number of data	Description
Normal	51677	Normal sample data
Fuzzers	1742	Vulnerability mining and analysis
Reconnaissance	4185	Penetration reconnaissance and collection
Backdoors	489	Backdoor injection attack
DoS	997	denial of service attack
Exploit	2638	malicious code
Analysis	298	Channel detection attack
Generic	205	Reconnaissance attack
Shellcode	231	Test channel, script attack
Worms	179	network vulnerability

Table 2: Parameter of ReXNet

Parameter	Value	Parameter	Value
Initialized learning rate	0.01	Optimizer function	Adam
Hidden layer in pooling	1	Classification loss function	cross entropy
Activation function	ReLU	dropout rate	0.5

Table 3: Comparison with different methods

Method	P	R	F1
MLDL	0.954	0.963	0.959
IMIDS	0.968	0.959	0.962
CGOA	0.975	0.965	0.971
Proposed	0.982	0.985	0.984

In order to further verify the robustness of the ReXNet model, comparative experiments are also conducted on the security data set CIC-IDS-2017, and the results are shown in Table 4. This new method performs well in accuracy, recall and F1 score. Most of the accuracy and recall values are around 0.97, and the F1 value is above 0.97.

Table 4: P, R and F1 on the CIC-IDS-2017 data set

Type	P	R	F1
Benign	0.978	0.976	0.977
Bot	0.953	0.977	0.965
DoS	0.980	0.946	0.978
DDoS	0.968	0.983	0.975

In this paper, we compare different types of log sets to verify the performance of intrusion detection model in the data fusion phase of multi-source logs using rule reasoning. Based on the ReXNet model mentioned above, a comparison experiment is conducted between rule-free inference and other feature fusion techniques. Since rule reasoning can distinguish uncertainty and unknown factors, the improved rule reasoning firstly extracts and refines information and fuses security events. On this basis, the basic probability distribution value of the information is given. The multi-source events from different network devices are preprocessed, simplified and introduced into different levels of confidence, and multiple attributes are fused for quantitative evaluation of security events to achieve the purpose of reducing the false positive rate. The experimental results are shown in Table 5.

5 Conclusions

In this paper, an intelligent cooperative robot data fusion technology based on ReXNet and rule reasoning is proposed to solve the problems of single evaluation information source and large accuracy deviation. By establishing ReXNet model based on attack mode of intelligent cooperative robots, feature learning and reconstruction of robot security data are carried out. Hierarchical mining of time-hidden features in data set is carried out. Parallel feature extraction is realized for multi-heterogeneous devices, which solves the shortcoming of insufficient feature extraction of existing decision level data. Finally, the confidence of classified data is deduced by using the data fusion method of rule inference. The experiment verifies the effectiveness of the overall model by using real data set and CIC-IDS-2017 data set. Compared with the existing algorithms, the robot intrusion detection model has better accuracy, and recall rate, and improves the classifi-

Table 5: P, R with different fusion models

Method	P(training)	P(test)	R(training)	R(test)
ReXNet-rule reasoning	0.974	0.948	0.976	0.959
ReXNet+rule reasoning	0.989	0.983	0.981	0.988

cation detection performance of intrusion network traffic. Through the analysis of multi-source heterogeneous logs of robots, accurate analysis of attack signals is realized and effective security rules are obtained, which is of great significance to the security intrusion detection and vulnerability detection of robots.

Acknowledgments

This work was supported by ”Horizontal project: Appearance design of intelligent cooperative robot SCR series”. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] I. Ahmad, Q. E. Ul Haq, M. Imran, *et al.*, “An efficient network intrusion detection and classification system,” *Mathematics*, vol. 10, no. 3, pp. 530, 2022.

[2] K. A. Dhanya, S. Vajipayajula, K. Srinivasan, *et al.*, “Detection of Network Attacks using Machine Learning and Deep Learning Models,” *Procedia Computer Science*, vol. 218, pp. 57-66, 2023.

[3] B. Hu, Z. H. Guan, F. L. Lewis, *et al.*, “Adaptive tracking control of cooperative robot manipulators with markovian switched couplings,” *IEEE Transactions on Industrial Electronics*, vol. 68, no. 3, pp. 2427-2436, 2020.

[4] L. Jiang, J. Shi, C. Wang, “Multi-ontology fusion and rule development to facilitate automated code compliance checking using BIM and rule-based reasoning,” *Advanced Engineering Informatics*, vol. 51, pp. 101449, 2022.

[5] S. Karim, G. Tong, J. Li, A. Qadir, U. Farooq, Y. Yu, “Current Advances and Future Perspectives of Image Fusion: A Comprehensive Review,” *Information Fusion*, vol. 90, pp. 185-217, 2023.

[6] K. H. Le, M. H. Nguyen, T. D. Tran, *et al.*, “IMIDS: An intelligent intrusion detection system against cyber threats in IoT,” *Electronics*, vol. 11, no. 4, pp. 524, 2022.

[7] S. Ma, Q. Zhang, T. Li, *et al.*, “Basic motion behavior recognition of single dairy cow based on improved Rexnet 3D network,” *Computers and Electronics in Agriculture*, vol. 194, pp. 106772, 2022.

[8] M. Naeem, T. Jamal, J. Diaz-Martinez, *et al.*, “Trends and future perspective challenges in big data,” in *Advances in Intelligent Data Analysis and Applications: Proceeding of the Sixth Euro-China Conference on Intelligent Data Analysis and Applications*, pp. 309-325, 2022.

[9] R. Nathan, C. T. Monk, R. Arlinghaus, *et al.*, “Big-data approaches lead to an increased understanding of the ecology of animal movement,” *Science*, vol. 375, 6582, eabg1780, 2022.

[10] C. Ounoughi, S. B. Yahia, “Data fusion for ITS: A systematic literature review,” *Information Fusion*, vol. 89, pp. 267-291, 2023.

[11] A. Ponmalar, V. Dhanakoti, “An intrusion detection approach using ensemble support vector machine based chaos game optimization algorithm in big data platform,” *Applied Soft Computing*, vol. 116, pp. 108295, 2022.

[12] M. Sandler, A. Howard, M. Zhu, *et al.*, “Mobilenetv2: Inverted residuals and linear bottlenecks,” *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510-4520, 2018.

[13] K. Shaheed, A. Mao, I. Qureshi, *et al.*, “Finger-vein presentation attack detection using depthwise separable convolution neural network,” *Expert Systems with Applications*, vol. 198, pp. 116786, 2022.

[14] J. R. Sun, M. S. Hwang, “A new investigation approach for tracing source IP in DDoS attack from proxy server”, in *Intelligent Systems and Applications*, pp. 850-857, 2015.

[15] M. Yang, I. Tjuawinata and K. Y. Lam, “K-Means Clustering With Local d-Privacy for Privacy-Preserving Data Analysis,” *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2524-2537, 2022.

[16] M. Yang, I. Tjuawinata, K. Y. Lam, J. Zhao and L. Sun, “Secure Hot Path Crowdsourcing With Local Differential Privacy Under Fog Computing Architecture,” *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 2188-2201.

[17] M. Yang, I. Tjuawinata, K. Y. Lam, T. Zhu and J. Zhao, “Differentially Private Distributed Frequency Estimation,” *IEEE Transactions on Dependable and Secure Computing*, 2022. doi: 10.1109/TDSC.2022.3227654.

Biography

Yu Jiang biography. Yu Jiang, Associate Professor, Shenyang Normal University. Research interests include data communication, interior design, security analysis.

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:

For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 7,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijns.jalaxy.com.tw or Email to ijns.publishing@gmail.com.